The Dissertation Committee for Paul Thomas Bauman II
certifies that this is the approved version of the following dissertation:

Adaptive Multiscale Modeling of Polymeric Materials Using Goal-Oriented Error Estimation, Arlequin Coupling, and Goals Algorithms

Committee:

J. Tinsley Oden, Supervisor

Leszek Demkowicz

Björn Engquist

Michael Marder

Mark Mear

C. Grant Willson

Adaptive Multiscale Modeling of Polymeric Materials Using Goal-Oriented Error Estimation, Arlequin Coupling, and Goals Algorithms

by

Paul Thomas Bauman II, B.S.; M.S.

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

May 2008

To the memory of my grandfather, and namesake, Paul Thomas Bauman I - I hope to someday come close to achieving his seemingly endless wisdom and strength of character.

To Mom and Steve, and Dad and Elaine for their unwavering love and support.

Most of all, to Sara, who managed to put up with all of it.

Acknowledgments

I would like to begin by expressing my sincere gratitude to my advisor, Professor J. Tinsley Oden. His patient guidance and relentless pursuit of excellence have helped shape me into a better scientist than I otherwise would have been. It has been an honor and a privilege working with him.

I also owe immense thanks to Dr. Serge Prudhomme who is possibly the most patient man I've ever met and has served as my co-mentor and friend throughout this work; our collaboration has been most fruitful.

I had the pleasure to work with Dr. David Littlefield as an undergraduate at the Institute for Advanced Technology. This work first was my first exposure to computational science and really solidified my choice to pursue graduate work in this field.

I would like to also thank Professor Leszek Demkowicz who first introduced me to the CAM program as an undergraduate. He has taught many excellent courses of which I had the pleasure to attend both as an undergraduate and a graduate student.

I would also like to acknowledge the staff at ICES, especially Lorraine Sanchez, Donna Larson, Sheila Townsend, and Sue Hennigar who have all made my stay much more pleasant and have always been very accommodating and understanding.

I appreciate very much the friends I've made in the CAM program both for their scientific insight and friendship, especially Chris Moore (and his wife Miranda), Mark Baumann, Chad Larson, Derek Gaston (and his wife Amanda), Jurijs Bazilevs (and his wife Anna), and David Fuentes. I have worked almost daily with David for the past 8 years and his never-ending sense of humor, kind demeanor, and dedicated work ethic have been a joy and an inspiration. I am very proud to call him one of my very best friends.

Many of us owe a great debt to the open source software community who have provided many tools that I now consider essential and have greatly enriched my work, including the developers of Linux, the GNU compilers, LaTeX, Subversion, and Xemacs. All of these packages have been used on daily basis during this work.

I would also like to thank the Department of Energy Computational Science Graduate Fellowship and the Krell Institute for providing financial support as well as many opportunities for scientific enrichment. Rachel Heidenreich at Krell especially provided outstanding assistance throughout my four years on the fellowship.

Finally, and most of all, I wish to thank those to whom I've dedicated this work: my family. Mom and Steve, and Dad and Elaine have never strayed from providing the most positive support, even when I don't call home or come visit all that often. My beautiful and loving wife, Sara, has managed for the past six years to forgive my many sleepless nights away from home, my occasional short temper, and my need to isolate myself at times during this work. I will always be grateful.

PAUL THOMAS BAUMAN II

The University of Texas at Austin
May 2008

Adaptive Multiscale Modeling of Polymeric Materials Using Goal-Oriented Error Estimation, Arlequin Coupling, and Goals Algorithms

Publication No. ____________________

Paul Thomas Bauman II, Ph.D.
The University of Texas at Austin, 2008

Supervisor: J. Tinsley Oden

Scientific theories that explain how physical systems behave are described by mathematical models which provide the basis for computer simulations of events that occur in the physical universe. These models, being only mathematical characterizations of actual phenomena, are obviously subject to error because of the inherent limitations of all mathematical abstractions. In this work, new theory and methodologies are developed to quantify such modeling error in a special way that resolves a fundamental and standing issue: multiscale modeling, the development of models of events that transcend many spatial and temporal scales. Specifically, we devise the machinery for a posteriori estimates of relative modeling error between a model of fine scale and another of coarser scale, and we use this methodology as a general approach to multiscale problems. The target application is one of critical importance to nanomanufacturing: imprint lithography of semiconductor devices.

The development of numerical methods for multiscale modeling has become one of the most important areas of computational science. Technological developments in the manufacturing of semiconductors hinge upon the ability to understand physical phenomena from the nanoscale to the microscale and beyond. Predictive simulation tools are critical to the advancement of nanomanufacturing semiconductor devices. In principle, they can displace expensive experiments and testing and optimize the design of the manufacturing process. The development of such tools rest on the edge of contemporary methods and high-performance computing capabilities and is a major open problem in computational science.

In this dissertation, a molecular model is used to simulate the deformation of polymeric materials used in the fabrication of semiconductor devices. Algorithms are described which lead to a complex molecular model of polymer materials designed to produce an etch barrier, a critical component in imprint lithography approaches to semiconductor manufacturing. Each application of this so-called polymerization process leads to one realization of a lattice-type model of the polymer, a molecular statics model of enormous size and complexity. This is referred to as the base model for analyzing the deformation of the etch barrier, a critical feature of the manufacturing process. To reduce the size and complexity of this model, a sequence of coarser surrogate models is generated. These surrogates are the multiscale models critical to the successful computer simulation of the entire manufacturing process. The surrogate involves a combination of particle models, the molecular model of the polymer, and a coarse-scale model of the polymer as a nonlinear hyperelastic material. Coefficients for the nonlinear elastic continuum model are determined using numerical experiments on representative volume elements of the polymer model. Furthermore, a simple model of initial strain is incorporated in the continuum equations to model the inherit shrinking of the material.

A coupled particle and continuum model is constructed using a special algorithm designed to provide constraints on a region of overlap between the continuum and particle models. This coupled model is based on the so-called Arlequin method that was introduced in the context of coupling two continuum models with differing levels of discretization. It is shown that the Arlequin problem for the particle-to-continuum model is well posed in a one-dimensional setting involving linear harmonic springs coupled with a linearly elastic continuum. Several numerical examples are presented. Numerical experiments in three dimensions are also discussed in which the polymer model is coupled to a nonlinear elastic continuum.

Error estimates in local quantities of interest are constructed in order to estimate the modeling error due to the approximation of the particle model by the coupled multiscale surrogate model. The estimates of the error are computed by solving an auxiliary adjoint, or dual, problem that incorporates as data the quantity of interest or its derivatives. The solution of the adjoint problem indicates how the error in the approximation of the polymer model influences the error in the quantity of interest. The error in the quantity of interest represents the relative error between the value of the quantity evaluated for the base model, a quantity typically unavailable or intractable, and the value of the quantity of interest provided by the multiscale surrogate model. To estimate the error in the quantity of interest, a theorem is employed that establishes that the error coincides with the value of the residual functional acting on the adjoint solution plus a higher-order remainder. For each surrogate in a sequence of surrogates generated, the residual functional acting on various approximations of the adjoint is computed.

These error estimates are used to construct an adaptive algorithm whereby the model is adapted by supplying additional fine-scale data in certain subdomains in order to reduce the error in the quantity of interest. The adaptation algorithm involves partitioning the domain and selecting which subdomains are to use the particle model, the continuum model, and where the two overlap. When the algorithm identifies

that a region contributes a relatively large amount to the error in the quantity of interest, it is scheduled for refinement by switching the model for that region to the particle model. Numerical experiments on several configurations representative of nano-features in semiconductor device fabrication demonstrate the effectiveness of the error estimate in controlling the modeling error as well as the ability of the adaptive algorithm to reduce the error in the quantity of interest.

There are two major conclusions of this study: 1. an effective and well posed multiscale model that couples particle and continuum models can be constructed as a surrogate to molecular statics models of polymer networks and 2. an error estimate of the modeling error for such systems can be estimated with sufficient accuracy to provide the basis for very effective multiscale modeling procedures. The methodology developed in this study provides a general approach to multiscale modeling. The computational procedures, computer codes, and results could provide a powerful tool in understanding, designing, and optimizing an important class of semiconductor-manufacturing processes.

The study in this dissertation involves all three components of the CAM graduate program requirements: Area A, Applicable Mathematics; Area B, Numerical Analysis and Scientific Computation; and Area C, Mathematical Modeling and Applications. The multiscale modeling approach developed here is based on the construction of continuum surrogates and coupling them to molecular statics models of polymer as well as *a posteriori* estimates of error and their adaptive control. A detailed mathematical analysis is provided for the Arlequin method in the context of coupling particle and continuum models for a class of one-dimensional model problems. Algorithms are described and implemented that solve the adaptive, nonlinear problem proposed in the multiscale surrogate problem. Large scale, parallel computations for the base model are also shown. Finally, detailed studies of models relevant to applications to semiconductor manufacturing are presented.

Contents

List of Tables

List of Figures

Chapter 1

Preliminaries

1.1 Introduction

The development of numerical methods for multiscale modeling has become one of the most important areas of computational science. Fundamental technological areas in which multiscale modeling is critical is in the manufacturing of semiconductors, which hinges upon the ability to understand physical phenomena from the nanoscale to the microscale and beyond. Indeed, by 2018, feature sizes of less than 15 nm are sought for manufacturing of production devices; the current production value is 45 nm. It is estimated that, using current techniques, the cost of manufacturing machinery that produce devices with such small features could reach as high as $65 million. This manufacturing cost is prohibitive to the industry. Thus, other processes must be devised to produce the desired technological advancements. Predictive simulation tools are critical to this advancement, to aid in the design of expensive experiments and testing, to assist in obtaining information where experimentation is simply not possible, and to optimize the design of the manufacturing process. Nevertheless, simulations of engineering systems within this context reach beyond the capability of current computing systems, making the dimensional reduction inherent in multiscale modeling an essential tool. The development of such simulation tools rests on the edge of contemporary modeling methods, high-performance computing capabilities, and is a major open problem in multiscale modeling.

In this dissertation, a molecular model is used to simulate the deformation of polymeric materials used in the fabrication of semiconductor devices. The goal is to develop a surrogate model that incorporates local fine scale information, where needed, and only coarse scale information in the remainder of the material body. This development requires not only the determination of a compatible coarse scale model from the fine scale model, but also the construction of an efficient, stable, and robust method to achieve coupling between the two.

Furthermore, the error in local quantities of interest is estimated using the framework of so-called goal-oriented error estimation. The error estimates can then be used to drive an adaptive *modeling* algorithm where the location and extent of the fine scale model is chosen such that the error in the quantity of interest is within a preset tolerance. These are the *Goals* algorithms. The remainder of this chapter introduces the theory of goal-oriented error estimation with the context of molecular statics, presents the major concepts involved in the next generation of lithography processes, followed by a literature survey of current techniques in multiscale modeling and previous work, and concludes with a discussion of the organization of the dissertation.

1.2 Theory of Goal-Oriented Error Estimation

1.2.1 Base Models of Molecular Statics

The concept of estimating and controlling modeling error in complex models of physical phenomena was advanced in earlier works [50, 52, 58, 53, 73]. More discussion of this literature is given in Section 1.4. The idea begins with the identification of a base model of the phenomena, characterized by the so-called *primal base problem*,

$$\boxed{\begin{array}{l}\text{Find } \mathbf{u} \in U \text{ such that} \\ \qquad B(\mathbf{u};\mathbf{v}) = F(\mathbf{v}) \qquad \forall \mathbf{v} \in V\end{array}} \tag{1.1}$$

where U and V are appropriate spaces of trial and test vectors, $B(\cdot;\cdot)$ is a semilinear form on $U \times V$, and F is a bounded linear functional on V. The semicolon denotes a possible nonlinear dependence of $B(\cdot;\cdot)$ on the entry $\mathbf{u}$ to the left of the semicolon, and linear dependence on $\mathbf{v}$.[1]

An example of (1.1) relevant to the current work is the nonlinear system of

---

[1] Equation (1.1) is obviously equivalent to the abstract problem $A\mathbf{u} = F$ in V', where A is an operator mapping U into the dual V' of V and $B(\mathbf{u};\mathbf{v}) - F(\mathbf{v}) = \langle A\mathbf{u} - F, \mathbf{v}\rangle$ for every $\mathbf{v}$ in V.

algebraic equations characterizing equilibrium configurations of systems of N particles (atoms or molecules):

$$\begin{aligned} E(\mathbf{u}) &= \min_{\mathbf{v}\in U} E(\mathbf{v}) \\ E(\mathbf{v}) &= \sum_{i=1}^{N} E_i(\mathbf{v}) - \sum_{i=1}^{N} \mathbf{f}_i \cdot \mathbf{v}_i \\ \mathbf{u}_i &= \mathbf{h}_i,\ i = N_0+1,\ldots,N \end{aligned} \tag{1.2}$$

Here, $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N)$ is the N-tuple of displacement vectors of the N molecules relative to a fixed reference configuration, E_i is the energy associated with molecule i, generally determined from inter-molecular energy potentials, $\mathbf{f}_i$ is the prescribed external force applied at site i, and $\mathbf{h}_i$ is a prescribed displacement on $N - N_0$ boundary sites, the displacement vectors $\mathbf{u}_i$ on N_0 interior molecules being unknowns. The setting is thus one in which the N atoms or molecules making up the model are initially located at points $\mathbf{x}_i$ in a bounded region $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega$. The displacements of $N - N_0$ points are prescribed, $\mathbf{u}_i = \mathbf{h}_i$, at points on $\partial\Omega$, while the displacements of the interior N_0 points are unknown, but constitute a minimizer of the total energy and correspond to an equilibrium configuration of the system. Then, in this case, the forms and spaces in (1.1) are

$$\begin{aligned} U &= \Big\{\Big\{\hat{\mathbf{h}}_i\Big\} + V; \hat{\mathbf{h}}_i = 0, i = 1,\ldots,N_0; \hat{\mathbf{h}}_i = \mathbf{h}_i, i = N_0+1,\ldots,N\Big\} \\ V &= \big\{\mathbf{v} = (\mathbf{v}_1,\ldots,\mathbf{v}_N) : \mathbf{v}_i = \mathbf{v}_i(\mathbf{x}_i), \mathbf{x}_i \in \overline{\Omega}; \mathbf{v}_i = 0, i = N_0+1,\ldots,N\big\} \\ B(\mathbf{u};\mathbf{v}) &= \sum_{i=1}^{N_0} \frac{\partial E_i(\mathbf{u})}{\partial \mathbf{u}} \cdot \mathbf{v}_i \\ F(\mathbf{v}) &= \sum_{i=1}^{N_0} \mathbf{f}_i \cdot \mathbf{v}_i \end{aligned} \tag{1.3}$$

The *goal* of this analysis is not merely the determination of the equilibrium N-vector $\mathbf{u}$ but the determination of a particular functional of $\mathbf{u}$ called the *quantity of interest*, which is characterized by a possibly nonlinear functional $Q : U \to \mathbb{R}$. The influence of the solution $\mathbf{u}$ to (1.1) on the quantity of interest $Q(\mathbf{u})$ is characterized by an N-vector $\mathbf{p} \in V$ which is a solution of the *dual or adjoint problem*,

$$\boxed{\begin{aligned} &\text{Find } \mathbf{p} \in V \text{ such that} \\ &\qquad B'(\mathbf{u}; \mathbf{v}, \mathbf{p}) = Q'(\mathbf{u}; \mathbf{v}) \qquad \forall \mathbf{v} \in V \end{aligned}} \tag{1.4}$$

where $\mathbf{u}$ is a solution of (1.1), $\mathbf{p} = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N)$, $\mathbf{p}_i = 0, i > N_0$, and, for $B(\cdot; \cdot)$ in (1.3),

$$\begin{aligned} B'(\mathbf{u}; \mathbf{v}, \mathbf{w}) &= \lim_{\theta \to 0} \theta^{-1} \left[B(\mathbf{u} + \theta \mathbf{w}; \mathbf{v}) - B(\mathbf{u}; \mathbf{v}) \right] \\ Q'(\mathbf{u}; \mathbf{v}) &= \lim_{\theta \to 0} \theta^{-1} \left[Q(\mathbf{u} + \theta \mathbf{v}) - Q(\mathbf{v}) \right] \end{aligned} \tag{1.5}$$

and, in the case that $(1.3)_3$ holds,

$$\boxed{B'(\mathbf{u}; \mathbf{v}, \mathbf{p}) = \sum_{i=1}^{N_0} \mathbf{v}_i \cdot \frac{\partial^2 E_i(\mathbf{u})}{\partial^2 \mathbf{u}_i} \cdot \mathbf{p}_i} \tag{1.6}$$

Note that the dual problem (1.4) is linear in $\mathbf{p}$.

1.2.2 Surrogate Models and Errors in Quantities of Interest

Let $(\mathbf{u}_0, \mathbf{p}_0)$ be an *arbitrary* pair of N-vectors in $U \times V$. Then, it is shown in [50] (see also [52]) that the error in the quantity of interest Q obtained by evaluating Q at $\mathbf{u}_0$ instead of at a solution $\mathbf{u}$ of (1.1) is

$$\boxed{\begin{aligned} \mathcal{E} &= Q(\mathbf{u}) - Q(\mathbf{u}_0) \\ &= \mathcal{R}(\mathbf{u}_0; \mathbf{p}) + \Delta \\ &= \mathcal{R}(\mathbf{u}_0; \mathbf{p}_0) + \mathcal{R}(\mathbf{u}_0; \mathbf{p} - \mathbf{p}_0) + \Delta \end{aligned}} \tag{1.7}$$

where $\mathcal{R}(\cdot; \cdot)$ is the residual functional,

$$\mathcal{R}(\mathbf{u}_0; \mathbf{v}) = F(\mathbf{v}) - B(\mathbf{u}_0; \mathbf{v}), \qquad \mathbf{v} \in V \tag{1.8}$$

and Δ is a remainder functional of higher order in the errors $\mathbf{e}_0 = \mathbf{u} - \mathbf{u}_0$ and $\varepsilon_0 = \mathbf{p} - \mathbf{p}_0$. The derivation of (1.7) assumes that the forms $B(\cdot; \cdot)$ and $Q(\cdot)$ are thrice differentiable in the sense of (1.5). The remainder Δ is then of the form

$$\begin{aligned}\Delta =& \frac{1}{2}\int_0^1 B''(\mathbf{u}_0 + s\mathbf{e}_0; \mathbf{e}_0, \mathbf{e}_0, \mathbf{p}_0 + s\varepsilon_0) - Q''(\mathbf{u}_0 + s\mathbf{e}_0; \mathbf{e}_0, \mathbf{e}_0)\, ds \\ &+ \frac{1}{2}\int_0^1 (Q'''(\mathbf{u}_0 + s\mathbf{e}_0; \mathbf{e}_0, \mathbf{e}_0, \mathbf{e}_0) - 3B''(\mathbf{u}_0 + s\mathbf{e}_0; \mathbf{e}_0, \mathbf{e}_0, \varepsilon_0) - \\ &\qquad B'''(\mathbf{u}_0 + s\mathbf{e}_0; \mathbf{e}_0, \mathbf{e}_0, \mathbf{e}_0, \mathbf{p}_0 + s\varepsilon_0))\,(s-1)s\, ds\end{aligned} \tag{1.9}$$

See [50].

The main ideas behind this approach to multi-scale modeling and estimating and controlling modeling error developed in this investigation are:

1. Choose $(\mathbf{u}_0, \mathbf{p}_0)$ "close" to the solution pair $(\mathbf{u}, \mathbf{p})$ so that the remainder Δ is negligible compared to $\mathcal{R}(\mathbf{u}_0; \mathbf{p})$.

2. A pair $(\mathbf{u}_0, \mathbf{p}_0)$ "close" to $(\mathbf{u}, \mathbf{p})$ can presumably be the solution of *surrogate* primal and dual problems of size $M \ll N$; e.g.

$$\boxed{\begin{aligned}&(\mathbf{u}_0, \mathbf{p}_0) \in U \times V : \\ &B_0(\mathbf{u}_0; \mathbf{v}) = F_0(\mathbf{v}) \qquad \forall \mathbf{v} \in V \\ &B_0'(\mathbf{u}_0; \mathbf{v}, \mathbf{p}) = Q'(\mathbf{u}_0; \mathbf{v}) \qquad \forall \mathbf{v} \in V\end{aligned}} \tag{1.10}$$

3. Generate a sequence of surrogate problems with solutions $(\mathbf{u}_0^k, \mathbf{p}_0^k)_{k\geq 1}$, so that for some integer m_0, the modeling error satisfies

$$\begin{aligned}|Q(\mathbf{u}) - Q(\mathbf{u}_0^k)| &\approx |\mathcal{R}(\mathbf{u}_0^k, \mathbf{p}_0^k) + \mathcal{R}(\mathbf{u}_0^k, \mathbf{p} - \mathbf{p}_0^k)| \\ &\leq \gamma_{tol}, \text{for } k > m_0\end{aligned} \tag{1.11}$$

where γ_{tol} is a preset error tolerance.

The implementation of these general ideas, of course, can present major challenges. To wit:

1. The generation of appropriate sequences of surrogates is the fundamental problem of multi-scale modeling. Surrogates can be based on coupling particle and continuum models, such as is the case in so-called atomistic-to-continuum approaches, or by dimensional reduction approaches such as methods based on the quasi-continuum methodologies [45].

2. The calculation of the solution $\mathbf{p}$ of the base dual problem, in general, represents an enormous and complex algebraic problem, even though it is linear. To avoid such calculations, one hopes to generate a sequence of surrogate dual solutions $\mathbf{p}_0^k$ that converges rapidly to $\mathbf{p}$ in V. Such sequences may be produced by *goals algorithms* or *goal-oriented adaptive schemes* [52, 58].

3. The modeling error estimate, $\mathcal{E} \approx \mathcal{R}(\mathbf{u}_0; \mathbf{p})$ involves the repeated evaluation of the full residual $\mathcal{R}(\mathbf{u}_0; \mathbf{p})$ at pairs $(\mathbf{u}_0^k, \mathbf{p}_0^k)$ of surrogate solutions. Approximations $\mathcal{R}^k(\cdot; \cdot)$ of the residual may be generated to reduce the cost of these calculations.

One major thrust of this work is the construction and implementation of such an algorithm and the characterization of its accuracy and efficiency.

1.3 Step and Flash Imprint Lithography

The principal application area of the multiscale modeling methods developed in this work involves modeling and simulation of molecular models of a manufacturing process called *imprint lithography* [7]. The primary goal of lithography[2] is to imprint features of computer chip components, such as wires, into a silicon substrate. This imprinting can be accomplished using a mask with the desired pattern in conjunction with photoresists and etching[3] procedures. In particular, optical lithography uses ultra-violet light shined through the masks to achieve the creation of the proper patterns. *Imprint lithography* proposes to use mechanical processes at room temperature to physically imprint the desired features. Imprint lithography is a recent development and is only beginning to reach commercialization. Nevertheless, it has already demonstrated capabilities of producing 32 nm features and a typical manufacturing unit costs \$5 million dollars [25], a substantial savings over the predicted optical lithography costs. A typical configuration used in imprint lithography consists of the following:

---

[2] The origin of the term lithography dates to 1796; it describes a process for transferring an image from a carved stone to paper [43]. In the current context, lithography refers to transferring a master pattern of an integrated circuit to a silicon wafer.

[3] The term "etching" refers to a process of transferring the pattern by chemical or physical removal of the underlying material [43].

- *The imprinting template.* The template is made of quartz crystal and has the desired features etched into its lower surface through an electron beam etching procedure. On the imprinted surface of the template is a release layer designed to allow the template to easily be removed from the polymerized etch barrier. The release layer is a perfluro alkane, very similar to Teflon.
- *The etch barrier solution.* The etch barrier undergoes chemical reactions when subjected to ultra-violet light. These chemical reactions form a solid polymer from a liquid monomer through a process called polymerization.
- *The transfer layer.* The transfer layer is a glassy styrene type of polymer used to "add aspect ratio" to the features imprinted in the etch barrier. In other words, the transfer layer increases the height-to-width ratio of the features to enhance the fidelity of the crucial, final etching procedures.
- *The substrate layer.* The substrate layer is generally made of silicon. Imprinting the desired features into the substrate is the primary goal of the lithography process.

The process of imprint lithography can be decomposed into eight stages [28]:

1. The liquid etch barrier solution is deposited in drops onto the transfer layer at several locations.
2. The template is placed on the surface so that the etch barrier solution fills the etched features on the template.
3. The sample is illuminated with ultra-violet light initiating the polymerization process thereby "solidifying" the features in the etch barrier.
4. The template is removed leaving the relief pattern.
5. An etch is performed to break through the residual etch barrier polymer between the features. This is the so-called breakthrough etch.
6. Another etch is performed, but now the goal is to etch the transfer layer so that the relief pattern is the same as the etch barrier, but the features have much larger aspect ratios.

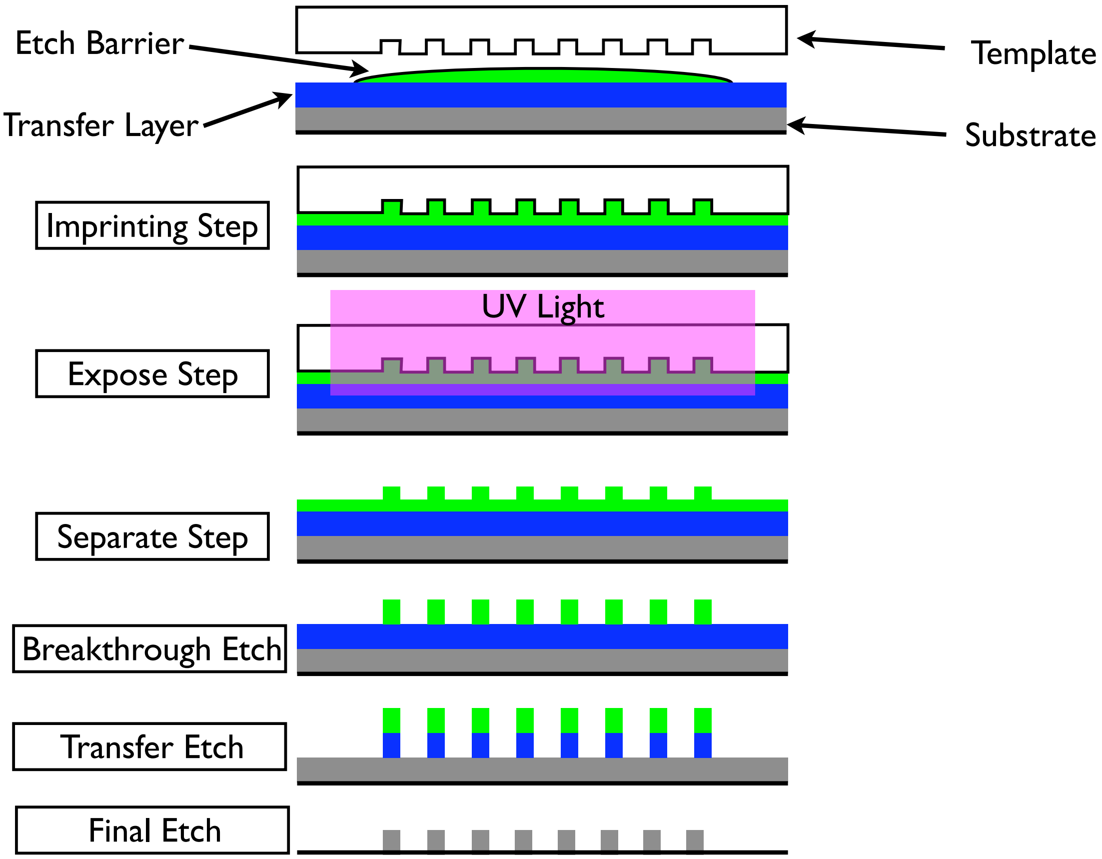


Figure 1.1: The Step and Flash Imprint Lithography (SFIL) process.

7. A final etch is performed to etch the feature pattern into the substrate.
8. Finally, the substrate is submerged in an organic solvent in order to remove residual polymeric materials from the substrate.

Figure 1.1 illustrates the entire process schematically. The key to success of this process is the fidelity of the features in the etch barrier prior to the breakthrough etch. The reactive ion etch that is used requires sharp features in order to produce a relief that mimics the template pattern. Appendix A gives a short discussion of the reactive ion etching process.

Thus, the goal of modeling the SFIL process is to study the deformation of the features of the etch barrier solution following the polymerization process. Furthermore, the error in certain quantities that reflect the deformation of the etch barrier will be quantified and controlled using the theory of goal-oriented error estimation and adaptive control, to be described in later chapters.

1.4 Literature Survey

The subject of multiscale modeling has been identified as one of the grand challenges of simulation-based engineering and computational mathematics [1]. There is a growing literature on this subject, but it is very much in its early stages of development. Here, relevant papers on coupling particle and continuum models in a solid mechanics context as well as literature related to adaptive modeling are discussed. Many other multiscale methods have been proposed for a wide variety of scientific disciplines. Few of these are based on rigorous mathematical arguments. For a representative survey, see [32].

Several review papers have been published on multiscale methods. The work of Liu *et al* [42] provides not only an overview of available methods but also an introduction to fundamental background information as well as an extensive list of references. Also, the work of Curtin and Miller [26] gives particular emphasis on methods related to the quasicontinuum method, and the work of E and co-workers [32] provides an overview of many types of existing multiscale methods, giving special attention to the heterogeneous multi-scale method.

The quasicontinuum method (QCM) [65, 45], developed originally by Tadmor [69] and extended by Ortiz, and Phillips [71], provides a systematic approximation of the molecular statics problem with a finite-element-based interpolation method. There are two key approximations in the QCM. First, the displacements of atoms removed far from a material defect are computed with finite element interpolation of so-called “rep-atoms”. The second approximation involves the calculation of the total energy: either a local calculation is performed in an element using the Cauchy-Born rule, or a nonlocal computation is carried out incorporating the full complexity of the atomistic model. The QCM has been applied to problems in fracture [44, 47], grain boundary interaction [64], and nanoindentation [70, 67]. An extension to dynamic problems is proposed in [29].

The Macroscale Atomistic *ab initio* Dynamics (MAAD) method or the Coupling of Length Scales (CLS) method, introduced by Broughton and co-workers [21], is often attributed to pioneering the dynamic coupling of quantum, molecular, and continuum scales. Concurrent coupling is achieved through “pad”, “overlap” or, “handshake” regions at the interfaces. The primary application of the MAAD method has been the brittle fracture of silicon [2].

The Coupled Atomistic and Discrete Dislocation plasticity (CADD) methodology [66] introduced by Shilkrot, Miller, and Curtin, also uses handshake regions to interface a two-dimensional molecular statics model with a continuum model. The continuum is linearly elastic, but with discrete dislocations introduced following the work of van der Giessen and Needleman [36]. These dislocations are transferred between the continuum and atomistic regions via the passing rules established in [66]. The motivation for this work stemmed from shortcomings of the QCM in that as dislocations formed and moved, the full atomistic model was used to resolve the dislocations, adding many nonessential degrees of freedom. The transfer of dislocations in the CADD method alleviates this issue. Extensions to three-dimensional and dynamic problems are to be addressed [66].

Wagner and Liu proposed the bridging-scale method [74] where a molecular dynamics model is coupled to a coarser model through a scale decomposition. More specifically, a region of the domain is modeled using a molecular dynamics model (fine scale) while the coarser scale model is solved over the entire domain. Equations of motion can be derived for the coarse scale by decomposing the displacements into mean (coarse) components and fine fluctuations. Using linearization of the forces at the interface, this decomposition allows the incorporation of (unknown) fine scale information from the coarse scale domain. The transferred information acts as a damping term for spurious wave reflection at the interface of the molecular dynamics and coarse scale models. This method has a natural decoupling of spatial and temporal scales so that the coarse scale dynamics does not proceed with the same time step as the molecular dynamics model. The method is applied to the analysis of carbon nanotubes in [60].

The heterogeneous multi-scale method (HMM) [30], first introduced by E and Engquist, is not a specific method, but rather a framework for designing methods, as stated by E *et al.* in [31]. The framework of the HMM can be described in two steps. First, one must choose the macroscale model and approximations. Step two is to choose an appropriate microscale model. In the case in which the macroscale model is invalid in some local region, the microscale model is solved locally. For the case in which the macroscale model is missing data, for example, constants for a constitutive equation are obtained from a simulation of the microscale model which is performed to acquire the necessary data for solving the macroscale model. The HMM has been applied to a number of mathematical problems in [30]. Furthermore, some *a priori* error estimates have been derived in [33] for elliptic problems.

Perhaps most relevant to the current work on multiscale coupling is the Arlequin method of Ben Dhia and co-workers [15, 16, 17, 18]. Although initially proposed in the context of coupling two continuum models, it is demonstrated in this work that the idea extends directly to coupling particle and continuum models. In the Arlequin method, two models share an overlap region, as in the MAAD method, but the deformations are constrained to match, in some sense, using Lagrange multipliers on the overlap. Furthermore, convex combinations of the energies are defined on the overlap with appropriate weight functions. The work of Belytschko and Xiao [14, 75] can be viewed as a special case of the Arlequin method. Guidault and Belytschko [37] have also investigated the coupling of two continuum models through numerical experiments. More recently, Fish *et al.* [35] and Badia and co-workers [6] have explored the notion of blending functions on an overlap region.

While each of the methods discussed heretofore have displayed apparent success in special model applications, many aspects of multi-scale modeling remain to be addressed. First, many of the schemes do not invoke estimation or control of error due to model approximation, and few use local error estimates; a notable exception is the work of Arndt and Luskin [3, 4]. Second, many of the methods lack decision criteria for application of the fine scale model – the local regions are chosen heuristically by the analyst. This is also the case for size determination of the overlap regions. It is important to be precise when determining the overlap region since, for example, as reported in [75], spurious wave reflection in dynamic problems can be generated. Next, there is little work in modeling polymer materials in this manner – all references cited here study homogeneous crystalline materials. Finally, the rigorous determination of compatible, coarse scale models is a relatively unaddressed issue as the choice of the coarse scale model is left to the analyst or is chosen for a specific application.

Initial efforts towards adaptive modeling by Oden, Zohdi and co-workers [76, 55, 54] used global error estimates. Later, following the work of Oden and Prudhomme on local error estimation in finite element methods [59, 49] and the work of Rannacher and colleagues [13, 11], Oden and Vemaganti successfully employed local error estimates based upon modeling error in quantities of interest [53, 73] to adaptively select models in heterogeneous elastic materials. This was accomplished using so-called Goals algorithms by partitioning the computational domain and selecting regions where the homogenized model was solved while in the remaining regions (those that are predicted to affect mostly the quantity of interest), the full heterogeneous model was solved. The mathematical formulation was later generalized to an abstract mathematical setting in [50] while the Goals algorithms have recently been extended in [52]. Recent efforts have extended this framework to wave propagation in elastic materials [62], materials with random coefficients [63], and coupling lattice and continuum models [51].

1.5 Previous Work

Previous work in adaptive error control for molecular statics with applications to crystalline materials is described in [58]. This work provided the foundation for the current development related to polymers. It is briefly summarized below. Note that mathematical rigor is neglected in this discussion for the sake of brevity; however, the reader is referred to [58] for full mathematical details.

1.5.1 Adaptive Error Control for Molecular Statics of Crystals

In [58], a thin film of aluminum crystal is indented by a rigid rectangular indenter, infinite in the out-of-plane direction, as depicted in Fig. 1.2. The dimensions for the block of crystal are 2000 × 1000 A^2 (A=Angström) in the [111] and [$\bar{1}$10] directions of the crystal. The crystal rests on a rigid support so that homogeneous boundary conditions $\boldsymbol{u}_i = \mathbf{0}$ are prescribed for those atoms i located at $y = 0$. The remaining boundary conditions are enforced as follows: at $x = 0$ and $x = 2000$, homogeneous essential boundary conditions are prescribed in the x- and z-direction while zero forces are prescribed in the y-direction; these boundary conditions enforce symmetry across the planes. The atoms in the $y = 1000$ plane (excluding the atoms under the indenter) are prescribed zero forces. The indenter is moved downward by a succession of increments $\delta l = 0.2$ A so that the boundary conditions for the atoms i just below the indenter are given by:

$$\boldsymbol{u}_i = (0, -s\,\delta l, 0), \qquad s = 1, \dots, 30 \tag{1.12}$$

Quasistatic steps are then considered to solve for the displacements.

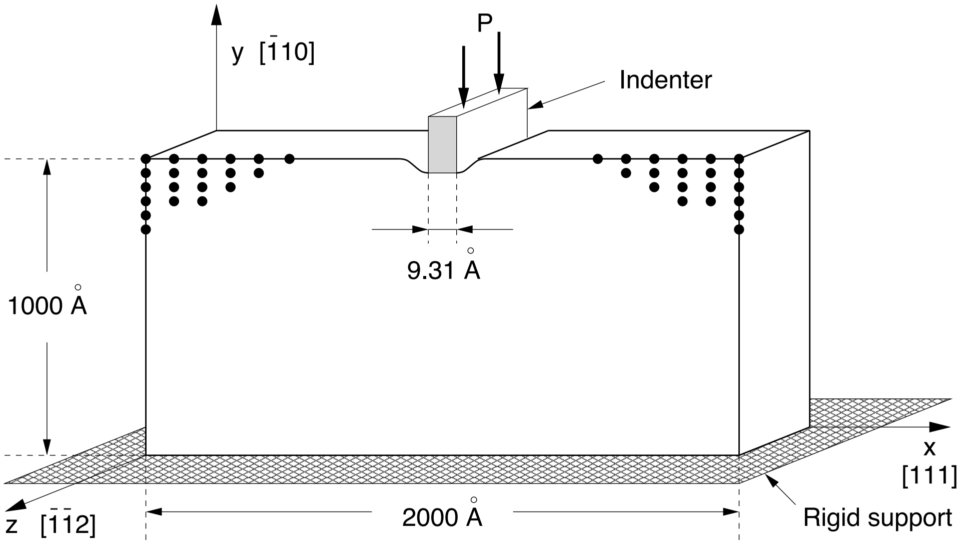


Figure 1.2: Nanoindentation of an aluminum crystal.

The indenter system is studied using molecular statics, as in (1.3), for the base model and the quasicontinuum method (QC) as a surrogate. As discussed in Section 1.4, the surrogate model consists of selecting a group of *representative atoms* (rep-atoms) for which the displacements are computed while the remaining atoms are linearly interpolated on a finite element mesh formed by the rep-atoms. This is stated formally as:

$$E_0(\boldsymbol{u}_0) = -\sum_{r=1}^{R} n_r\, \boldsymbol{f}_{0,r} \cdot \boldsymbol{u}_{0,r} + \sum_{r=1}^{R} n_r E_r(\boldsymbol{u}_0) \tag{1.13}$$

where $\boldsymbol{u}_0$ is the displacement of the rep-atoms, $E_r(\boldsymbol{u}_0)$ is the energy computed for each rep-atom (this can be local or nonlocal depending upon the rep-atom), n_r is a weighting coefficient to account for the number of atoms contained within an element formed by the rep-atoms, $\boldsymbol{f}_{0,r}$ is an applied load to a rep-atom r, and R is the number of rep-atoms. A solution is computed with sufficiently many rep-atoms to be considered "exact" for the purposes of error comparison. Figure 1.3 shows the mesh for this "overkill" solution as well as an example of a QC mesh.

The Goals algorithms are implemented within the QC code [46]. The quality of the error estimator and the meshes generated is studied. Of particular interest is the construction of the error estimator. Let π denote the extension of the current lattice (rep-atoms) to the full lattice space. It is well known that the contribution $\mathcal{R}(\boldsymbol{\pi}\boldsymbol{u}_0; \boldsymbol{\pi}_0\boldsymbol{p}_0)$ vanishes for Galerkin approximations. In a similar manner here, this

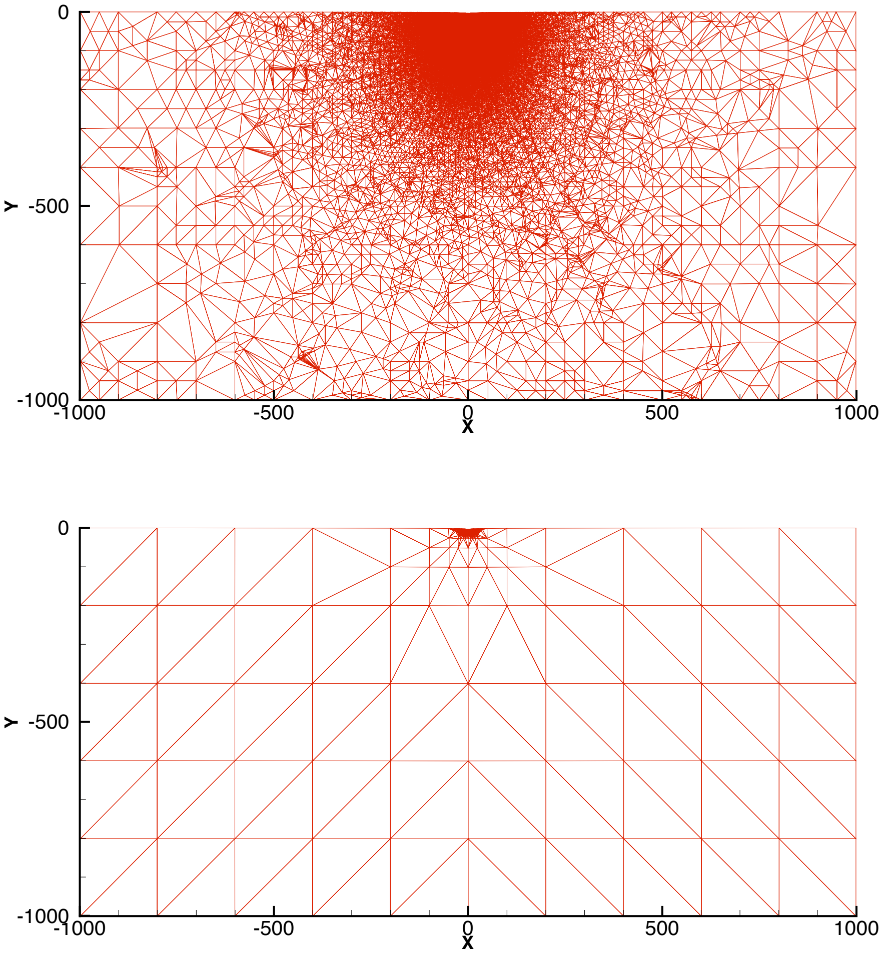


Figure 1.3: Finite element triangulation for the base model solution (top) and the QC solution (bottom) at load step 15. The base model solution has 25,484 atoms while the QC solution has 445 atoms.

term fails to detect the model reduction error; that is, solving the adjoint on the same lattice as the primal problem and then evaluating the residual does not capture any error. It follows that the solution $\boldsymbol{p}_0$ provides an unacceptable approximation of the dual solution $\mathbf{p}$, in the sense that $\mathcal{R}(\boldsymbol{\pi}\boldsymbol{u}_0;\boldsymbol{\pi}_0\boldsymbol{p}_0)$ approximates $\mathcal{R}(\boldsymbol{\pi}\boldsymbol{u}_0;\mathbf{p})$ poorly, and a better approximation should be obtained.

The strategy is to "enrich" the lattice by adding more repatoms to the lattice, but still many fewer than the full lattice, solving the adjoint problem, and evaluating the residual on this enriched configuration. Quantities on this configuration are labeled $\tilde{(\cdot)}$. Thus, $\tilde{\boldsymbol{\pi}}\mathbf{u}_0$ extends the displacement vector from R rep-atoms to $\tilde{N}$ atoms. Then, the following adjoint problem is solved:

$$\begin{aligned}&\textit{Find } \tilde{\mathbf{p}} \in \tilde{V}_0 \textit{ such that}\\ &\qquad\qquad \tilde{B}'(\tilde{\boldsymbol{\pi}}\boldsymbol{u}_0;\tilde{\mathbf{v}},\tilde{\mathbf{p}}) = \tilde{Q}'(\tilde{\boldsymbol{\pi}}\boldsymbol{u}_0;\tilde{\mathbf{v}}), \qquad \forall \tilde{\mathbf{v}} \in \tilde{V}_0\end{aligned} \tag{1.14}$$

where $\tilde{V}_0$ represents the space of vectors of length $\tilde{N}$ with zero Dirichlet conditions beneath the indenter. Furthermore, the residual is then computed as

$$\tilde{\mathcal{R}}(\tilde{\mathbf{u}};\tilde{\mathbf{v}}) = \sum_{i=1}^{\tilde{N}_a} \tilde{\boldsymbol{r}}_i(\tilde{\mathbf{u}}) \cdot \tilde{\mathbf{v}}_i \tag{1.15}$$

where $\tilde{\boldsymbol{r}}_i = \tilde{\boldsymbol{f}}_i - \left[\frac{\partial E_i}{\partial \boldsymbol{u}}(\tilde{\boldsymbol{\pi}}\boldsymbol{u}_0)\right]$. In the examples here, the quantity of interest is the force on the impinging indenter:

$$Q(\mathbf{u}) = -\sum_{i=1}^{M} \frac{\partial E_i}{\partial \boldsymbol{u}}(\mathbf{u}) \cdot \boldsymbol{n} \tag{1.16}$$

Figure 1.4 illustrates the effectiveness of the error estimate discussed previously. With one exception, the error estimate produces quite accurate results and is often observed to be an upper bound of the true error, although this is not necessarily the case.

Finally, this error estimate is partitioned amongst the elements and used to drive an adaptive algorithm. Figure 1.5 shows the force-displacement curves generated by the base model (overkill) solution, the quasicontinuum method, and the Goals algorithms. While the quasicontinuum solution may be acceptable, the Goals solution produces a much more accurate representation of the onset of dislocations, while still retaining quite acceptable computational cost. Furthermore, Figure 1.6 compares

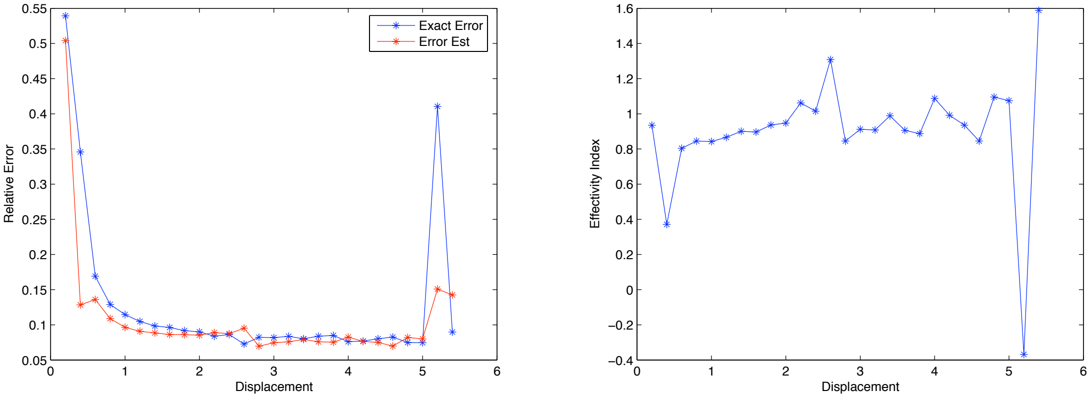


Figure 1.4: The relative error (left) and the effectivity indices (right) are shown comparing the error estimator to the exact error for the enriched QCM mesh.

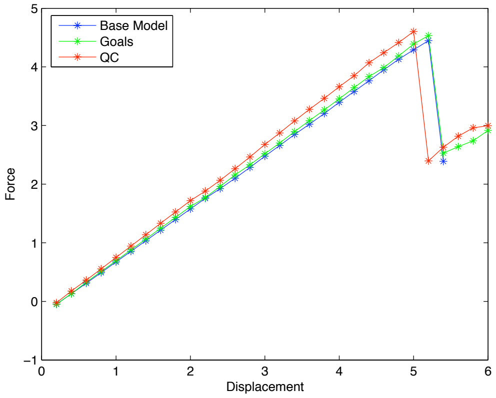


Figure 1.5: Force-displacement curves computed from the base model solution, the QC solution, and the Goals solution.

the meshes generated by the quasicontinuum algorithm and the Goals algorithm. Although the Goals algorithm calls for approximately double the number of atoms in the solution, the mesh enlarges much more smoothly away from the defect, and is generated using an error estimate derived from specific rational arguments.

1.6 Summary and Organization of Dissertation

In this Introduction, the theoretical framework for goal-oriented error estimation is established for problems in molecular statics. This includes general statements of the

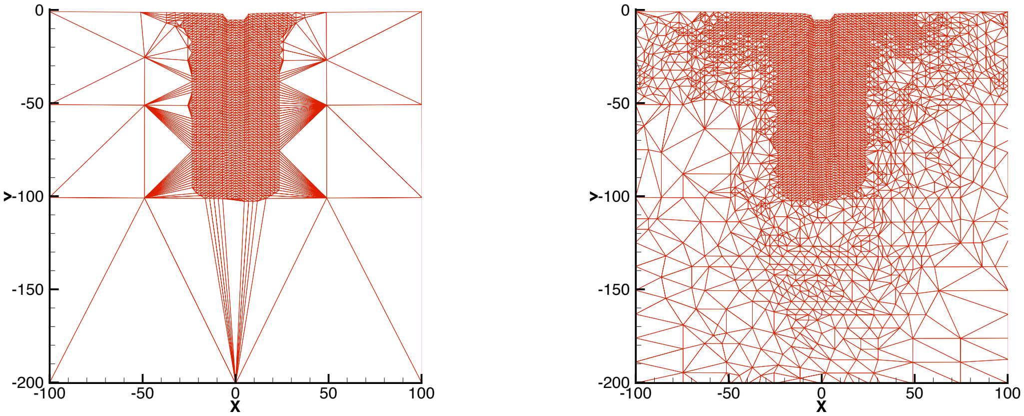


Figure 1.6: QCM (left) and Goals (right) mesh at load step 27. The number of atoms in the QCM and Goals meshes are 1,629 and 3,452, respectively.

primal and adjoint problems for both the base model and the surrogate model as well as the residual-based error estimate where the surrogate primal solution is weighted by the solution of the adjoint problem to estimate the error in a quantity of interest. This sets the stage for the construction of surrogate models that couple particle and continuum models and for estimating the error incurred in such an approximation. Next, the Step and Flash Imprint Lithography technique is introduced and briefly described. The focus of the modeling and simulation efforts presented in this work is on developing tools to accurately and effectively simulate critical stages in the lithography process. The current literature on multiscale methods and technology is surveyed for techniques related to the present work. Finally, a brief summary on a study in adaptive multiscale modeling for crystalline materials using the quasicontinuum method is given; the quasicontinuum investigation is the basis for the present work on polymeric materials.

The remainder of the dissertation is organized as follows. Chapter 2 presents the detailed construction of the molecular base model: a description of the polymerization process and algorithm used to simulate the chemistry of polymerization, the choice of potentials used in the resulting molecular statics model, and a discussion of the algorithms used to solve the nonlinear optimization problem. Some numerical examples are presented to illustrate typical solutions of the base model as well as convergence behavior of the solutions.

Chapter 3 introduces an algorithm for constructing consistent, continuum models that approximate the coarse-scale behavior of the molecular model. In particular, a technique is developed where unknown parameters in the chosen constitutive equations for nonlinearly elastic materials are determined by numerical experiments on representative volume elements of the polymer base model. Furthermore, the finite element formulation used in the solution of the resulting continuum model is augmented in order to incorporate the inherent "shrinkage" behavior observed in the lattice model. Numerical examples are given illustrating both the experiments on representative volume elements and the results of the new continuum formulation.

Chapter 4 develops a new formulation for the coupling of particle and continuum models that is based on using Lagrange multipliers that enforce constraints on a region of overlap between the two models. A mathematical analysis is presented that demonstrates well-posedness on a model one-dimensional model that couples harmonic springs and an elastic bar. The formulation is extended to the nonlinear, three-dimensional case where the polymer model introduced in Chapter 2 is coupled to the continuum model developed in Chapter 3. Numerical examples illustrate the character of solutions and convergence behavior.

Chapter 5 applies the framework for goal-oriented error estimation established in this chapter to the surrogates developed in Chapter 4. In particular, a projection operator is discussed that maps the Arlequin solution to the full lattice space so that the adjoint problem is solved for the entire lattice. This same procedure is also used for the calculation of the residual. Several numerical experiments illustrate the fidelity of the error estimates. Next an adaptive modeling algorithm is constructed that is based on a partition of the residual error estimate. Again, numerical results demonstrate the effectiveness of the algorithm.

Chapter 6 introduces approximations to the solution of the dual problem and evaluation of the residual. This is necessary as, in principle, the adjoint and residual calculations on the full lattice space are cost prohibitive. Again, numerical examples demonstrate the effectiveness of the error estimator and adaptive modeling algorithm.

Finally, the dissertation is concluded in Chapter 7 with a summary of results and recommendations for continuation of the work presented here.

Chapter 2

The Base Model

2.1 Introduction

In this chapter, the details of the chemical analysis related to the etch barrier polymer are introduced along with a computational scheme designed to model the polymerization process. Once realizations of the polymer network are established through the polymerization process, a molecular statics model is created through a procedure that assigns molecular potentials to model the bonds between the polymer molecules. Finally, solution algorithms for the problem of solving very large systems of nonlinear molecular statics equations are discussed in detail, followed by numerical examples to illustrate the convergence properties.

2.2 The Polymerization Process and the Polymer Model

In the SFIL process, described in Section 1.3, the target etch barrier material is produced in two steps:

1. a solution of chemical constituents flows into the template relief (the template contours) as the template is pressed toward the transfer layer;
2. ultra-violet (UV) light is passed through the template (which is transparent quartz) into the mixture.

The light causes chemical reactions between the chemical constituents which leads to the formation of chain-like macromolecules consisting of various repeated molecular units thereby solidifying the liquid. The units are monomers and the macromolecules are polymer chains. The resulting material is a polymer and the chemical process is called polymerization. The process is outlined in this chapter and mathematical models and algorithms that define the chemical and mechanical properties of the etch-barrier polymer are presented for a representative set of constituents.

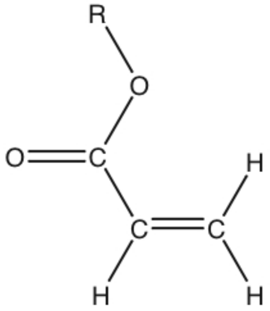
2.2.1 Initial monomer mixture

An acrylate-based[1] mixture is prepared which is composed of four molecular constituents: two distinct monomer molecules, a cross-linker molecule, and a photoinitiator molecule. The chemical structure of each molecule is described below. Note that, in what follows, the chemical structures as shown are not conventional. They are displayed in a structure that differentiates the role of each chemical constituent.

Monomer $\mathbf{M}_1$

One of the monomers is a silicon monoacrylate used for etch resistance.[2] An example is Gelest SIA-0210

$M_1 \sim$

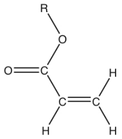


(2.1)

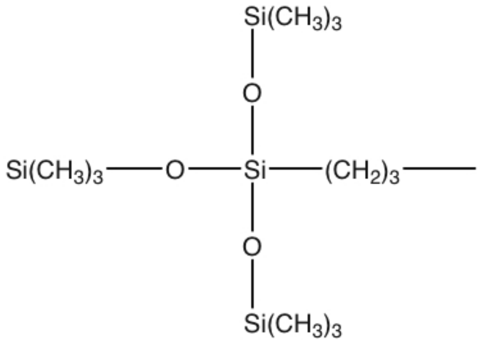
where

---

[1] An acrylate is a chemical compound that contains the functionality



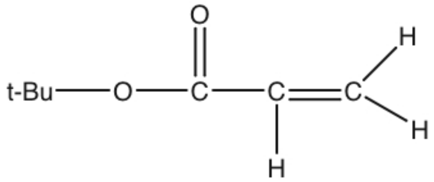
[2] The term "etch resistance" refers to the fact that the barrier can delay the etching process relative to the transfer layer.



$$R \sim \mathrm{Si(CH_3)_3{-}O{-}Si(O{-}Si(CH_3)_3)(O{-}Si(CH_3)_3){-}(CH_2)_3{-}} \tag{2.2}$$

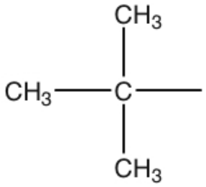
Monomer M_2

The second monomer is a reactive diluent designed to maintain low viscosity. One choice is the t-Butyl[3] acrylate (TBA) given by



$$M_2 \sim \mathrm{t\text{-}Bu{-}O{-}C({=}O){-}CH{=}CH_2} \tag{2.3}$$

where t-Bu stands for the subgroup



$$\text{t-Bu} \sim \mathrm{CH_3{-}C(CH_3)(CH_3){-}} \tag{2.4}$$

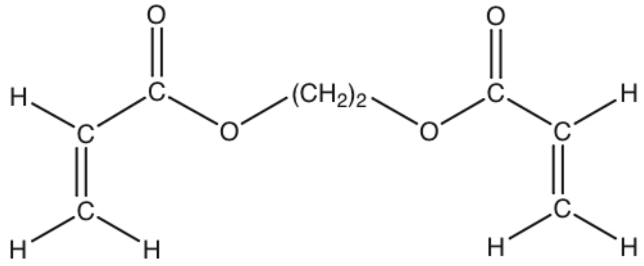
Cross-linker XL

A cross-linker molecule is added to provide mechanical stability and strength to the polymer network by cross-linking the main polymer chain, e.g. Ethylene Glycol Diacrylate (EGD):

---

[3] The reactant (2.4) is referred to as “t” Butyl because the carbon atoms form a “t”-structure and connect three CH_3 radicals, a characteristic of the butyl group [57].

XL $\sim$

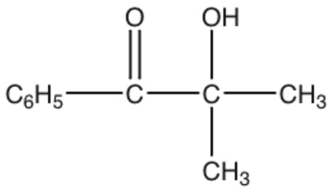


(2.5)

Photoinitiator

A highly reactive molecule is added that reacts to ultra-violet light exposure to initiate the polymerization process. The commercially-available photoinitiator Darocur 1173 is considered here:

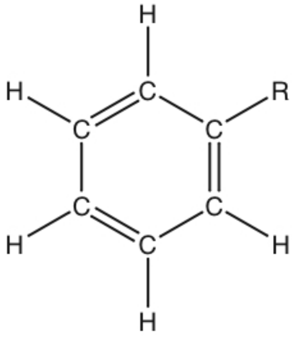
I $\sim$



$$C_6H_5-\overset{\overset{\displaystyle O}{\|}}{C}-\overset{\overset{\displaystyle OH}{|}}{\underset{\underset{\displaystyle CH_3}{|}}{C}}-CH_3 \qquad (2.6)$$

where C_6H_5 denotes the phenyl structure.

C_6H_5 $\sim$



(2.7)

2.2.2 Dispense step

The four-component solution is dispensed onto a flat polymeric transfer layer which has been bonded to a silicon wafer, the substrate. The substrate is held in place by a vacuum mechanism, and the quartz template is lowered into contact with the etch-barrier mixture, which fills the template relief; i.e. the mixture assumes the pattern of the template contact topography.

2.2.3 Light exposure and polymerization

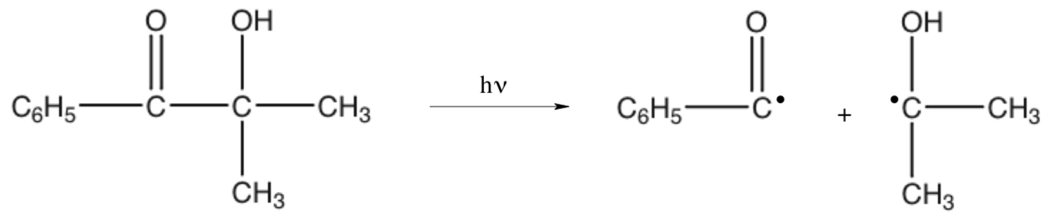
Next, ultra-violet light is applied to the template which causes the photoinitiator to break into two radical species, one more reactive than the other. This creates a rapid chemical process in which the monomers, cross-linkers, and radicals interact to form a polymeric material fitting the template cavities and resting on the transfer layer. The final molecular composition of the polymer depends upon these various chemical reactions.

The initiator I and light reaction is denoted

$$\mathrm{I} + h\nu \longrightarrow \mathrm{R}_1^\bullet + \mathrm{R}_2^\bullet \tag{2.8}$$

symbolizing that an initiator molecule reacts with photons supplied at an energy of $h\nu$, h being Planck's constant and ν the frequency of ultra-violet light, and produces the two free radicals $\mathrm{R}_1^\bullet$ and $\mathrm{R}_2^\bullet$. The "dot" here symbolizes the existence of an unpaired electron. An example of $\mathrm{R}_1^\bullet + \mathrm{R}_2^\bullet$ is given in Box 1.

Box 1: Schematic of Reaction (2.8)

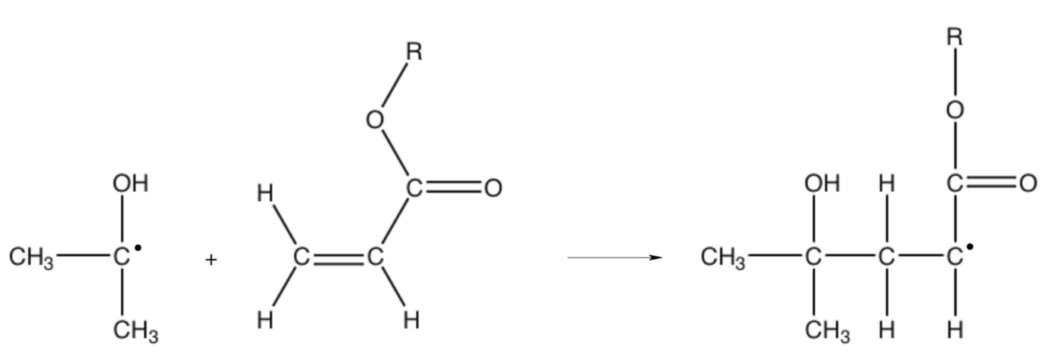


The free radicals react with a monomer or cross-linker in a process called *propagation*, symbolically represented as:

$$\begin{aligned} \mathrm{M} + \mathrm{R}^\bullet &\longrightarrow \mathrm{MR}^\bullet \\ \mathrm{M} + \mathrm{M}_n\mathrm{R}^\bullet &\longrightarrow \mathrm{M}_{n+1}\mathrm{R}^\bullet \end{aligned} \tag{2.9}$$

Thus, in the first reaction, a monomer (M_1, M_2, or XL) reacts with a radical species made available by the light-exposure reaction, to form a new radical $\mathrm{MR}^\bullet$. Examples are given in Box 2. Likewise, radicals of the type $\mathrm{M}_n\mathrm{R}^\bullet$, n being the number of monomers in the chain, can react with another monomer M to produce a new chain $\mathrm{M}_{n+1}\mathrm{R}^\bullet$.

Box 2: Schematic of Reaction (2.9)



Eventually, chain end radicals will recombine to finally create a polymer, in a process called *termination*, that can occur either as a *combination*,

$$\mathrm{M}_n\mathrm{R}^{\bullet} \; + \; \mathrm{M}_m\mathrm{R}^{\bullet} \longrightarrow \mathrm{R}\mathrm{M}_{n+m}\mathrm{R} \tag{2.10}$$

or *disproportionation*,

$$\mathrm{M}_n\mathrm{R}^{\bullet} \; + \; \mathrm{M}_m\mathrm{R}^{\bullet} \longrightarrow \mathrm{M}_n \; + \; \mathrm{M}_m \tag{2.11}$$

There is also a process called *radical scavenging* in which a free radical reacts with oxygen to create a stable peroxy radical[4]

$$\mathrm{R}^{\bullet} + \; \mathrm{O}_2 \longrightarrow \text{Peroxy Radical} \tag{2.12}$$

Dissolved oxygen in the solution can negatively influence the polymerization process. The radicals generated by the photoinitiators rapidly react with molecules of oxygen, forming the peroxy species, that do not react with acrylate double bond. Only when the oxygen is depleted do the radicals react with the monomers to form the growing polymer chains, which continue to propagate until they encounter a radical end of another chain, at which point there is termination by combination or disproportionation.

---

[4] A peroxy component is a component containing the bivalent group O_2. See [22].

2.2.4 Reaction rate equations

Fundamental to the characterization of the final polymer composition are the rate equations of chemical kinetics which are empirical relations between rates of initiation, radical scavenging, polymerization, and termination[5] [27]:

$$\begin{aligned} &\text{Rate of Initiation:} & R_I &= 2\varphi k_I[\mathrm{I}] && (2.13)\\ &\text{Rate of Oxygen Scavenging:} & R_{O_2} &= -2k_O[\mathrm{O_2}][\mathrm{R}] && (2.14)\\ &\text{Rate of Propagation:} & R_P &= -2k_P[\mathrm{M}][\mathrm{R}] && (2.15)\\ &\text{Rate of Termination:} & R_t &= -2k_t[\mathrm{R}][\mathrm{R}] && (2.16) \end{aligned}$$

Here, brackets [ ] indicate the concentration of species in moles per liter (mol/L), k_I is the reaction coefficient for initiation (in s^{-1}), k_O and k_p are the rate coefficients for oxygen consumption and polymerization (in L/(mol·s)), and k_t is the rate of termination coefficient. A discussion of fundamental concepts in chemical kinetics is given in Appendix B. In (2.13), φ is the quantum yield that is the product $\mu\varphi_m$, φ_m being the number of initiator molecules dissociated per photon, and μ the fraction of radicals produced that initiate the propagating polymer chains. The notation [R] represents the concentration of all radicals regardless of chain length, including for instance, $\mathrm{M}_n\mathrm{R}$.

Equations (2.13)-(2.16) are the empirical laws relating rates to measures of species concentrations. The reaction and rate coefficients must be determined experimentally and are critical to the accurate modeling of the polymerization process. Measured values of these coefficients are reported in [28].

2.2.5 Conservation of species

The species balance laws follow from the rate equations:

$$\begin{aligned} \frac{d[\mathrm{I}]}{dt} &= -k_I\varphi_m[\mathrm{I}]\\ \frac{d[\mathrm{R}]}{dt} &= -2k_I\varphi[\mathrm{I}] - k_O[\mathrm{O_2}][\mathrm{R}] - 2k_t[\mathrm{R}]^2 \qquad (2.17)\\ \frac{d[\mathrm{M}]}{dt} &= -k_p[\mathrm{M}][\mathrm{R}] \end{aligned}$$

---

[5] Following [27], disproportionation is ignored in this model.

Initial species concentrations, prior to the UV exposure, are known. The concentration $[O_2]$ of oxygen is presumed to be known, or can be estimated independently by solving a diffusion equation characterizing oxygen infiltration from the template boundaries.

2.2.6 Activation energies, rates, and probabilities

The minimal energy necessary for a specific chemical reaction to occur is called the *activation energy*, denoted E_a. Thus, E_a is the additional energy that must be absorbed by the reacting molecules beyond their ground states to allow them to pass through a high-energy transition state before products (stable molecules) are formed. Generally, the kinetic energy of molecules must be sufficient to push the ground state energy over the activation energy barrier to create a reaction, and this kinetic energy is proportional to the absolute temperature of the collection of molecules.

The classical Arrhenius law describes an empirical relationship between the activation energy, the absolute temperature T, and the reaction rate k:

$$k = Ae^{-E_a/\kappa T} \tag{2.18}$$

where A is a constant, called the frequency factor, which has the same units as the rate constant k, and κ is Boltzmann's constant. The factor $\exp(-E_a/\kappa T)$ is the Maxwell-Boltzmann distribution, giving the fraction of molecules that have kinetic energy greater than E_a. The Arrhenius law is known to be remarkably accurate for a wide range of chemical reactions [40].

The relationship between the reaction rate k of (2.18) and the probability P of a reaction is of fundamental importance. Since $\exp(-E_a/\kappa T)$ is the fraction of molecules with energy greater than E_a, and E_a is the threshold energy for a reaction to occur, the probability that a species will undergo a reaction is proportional to k:

$$P = Ce^{-E_a/\kappa T} \propto k \tag{2.19}$$

where C is such that P is normalized.

2.2.7 A kinetic Monte-Carlo process

The kinetic equations (2.17) only provide the global species concentrations as a function of time (and the initial species distributions). To determine the molecular structure at the termination of the polymerization process, one must follow the likely chemical reactions that can occur between the molecular components. The possible conformations resulting from reactions of a given initial distribution of constituents can be generated through a Monte-Carlo-like algorithm that shall be referred to here as the kinetic Monte-Carlo process. This process was introduced in [23] and discussed here for the sake of completeness.

The constituent lattice

A three-dimensional regular lattice $\mathcal{L}$ is introduced. The number of lattice sites is set equal to the estimated number of constituent molecules in a unit cube Ω of the initial etch barrier mixture. The lattice sites are merely addresses of molecular constituents that will be the basis of a bookkeeping scheme to trace the path of reactions during the polymerization process.

There are five constituents in the SFIL process under study: the monomer M_1, the monomer M_2, the cross-linker XL, the initiator I, and possible voids V. The voids are introduced to allow diffusion of the constituents during the model process. Let the concentrations by volume of each of these constituents before photo-curing (UV exposure) be denoted C_{M_1}, C_{M_2}, C_{XL}, C_I, and C_V, respectively. Then,

$$C_{M_1} + C_{M_2} + C_{XL} + C_I + C_V = 1.0 \tag{2.20}$$

Lattice cell assignment

Five disjoint subintervals of $I = [0, 1]$ are defined with length equal to the fraction of each constituent:

$$\begin{aligned}
I_{M_1} &= [0, C_{M_1}] \\
I_{M_2} &= [C_{M_1}, C_{M_1} + C_{M_2}] \\
I_{XL} &= [C_{M_1} + C_{M_2}, C_{M_1} + C_{M_2} + C_{XL}] \\
I_I &= [C_{M_1} + C_{M_2} + C_{XL}, C_{M_1} + C_{M_2} + C_{XL} + C_I] \\
I_V &= [1 - C_V, 1]
\end{aligned}$$

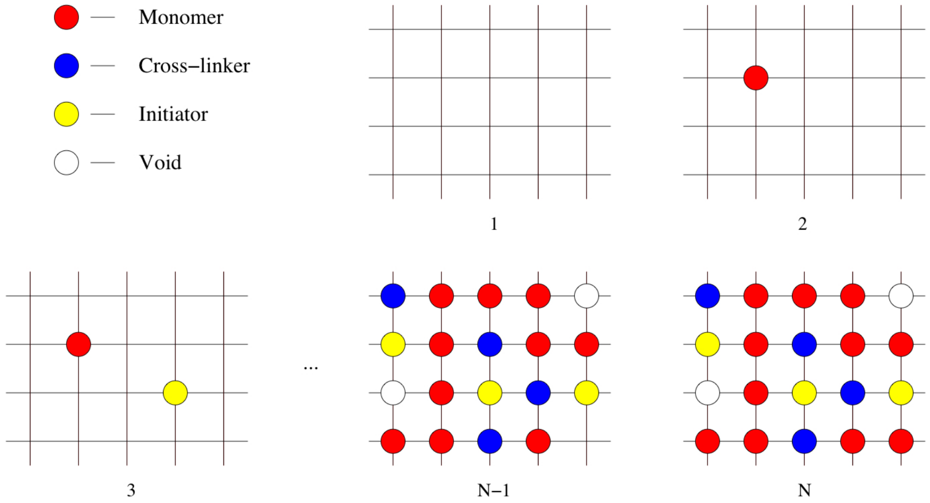


Figure 2.1: A schematic of the lattice cell placement algorithm.

Let N denote the number of cells in $\mathcal{L}$ and let j be an index which specifies the constituent types $j = \{\mathrm{M}_1, \mathrm{M}_2, \mathrm{XL}, \mathrm{I}, \mathrm{V}\}$. Then a molecular constituent is assigned to each cell as follows.

1. Boundary cells are assigned either a template molecule or a transfer layer molecule, depending on the location of the part of the boundary where the cell is located.
2. Each lattice site is visited in order and a uniform random number, r, is selected such that $0 \leq r \leq 1$. If $r \in I_j$, then the cell is assigned constituent j.
3. A random swapping procedure of the cells is used to further "randomize" the lattice.

Figure 2.1 illustrates the lattice placement process schematically.

Kinetic Monte-Carlo Algorithm

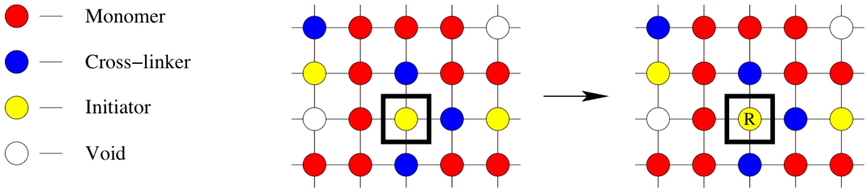
The following Monte-Carlo type algorithm performs the polymerization of the now populated lattice. The algorithm was introduced in [23] and is carefully reproduced here for completeness. Let N_c be the number of cycles, N_{c_i} be the number of iterations per cycle, and N_i the number of initiation cycles. Then, for each cycle C, an *initialization* loop and a *propagation* loop is executed.

1. Initialization Loop

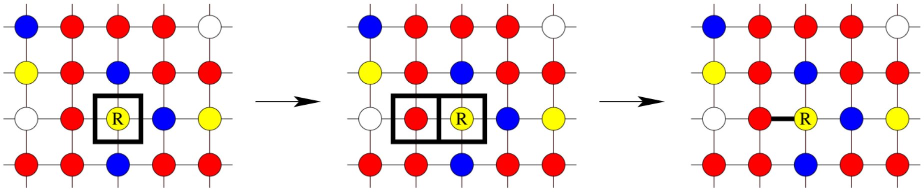
   (a) Select a random lattice site and check if the constituent randomly assigned to the cell is an initiator.
   (b) If the site is an intiator, select a uniform random number r, $0 \leq r \leq 1$. If $r \leq P_I$, P_I being the probability of an initiator reacting, the initiator reacts and the cell is now labeled a free radical R; otherwise the intiator does not react and the cell label is unchanged.
   (c) Repeat N_i times.

2. Propagation Loop
   (a) A random site is selected and the cell label j identified.
   (b) If j is a void, then no reaction occurs and the loop is incremented.
   (c) If j is an intiator, a random number r, $0 \leq r \leq 1$, is selected. If $r \leq P_I$, a reaction occurs; otherwise no reaction occurs.
   (d) If j is a radical (or a polymer chain with an active radical), a random neighbor is selected, where a neighbor is defined as an adjacent site in one of the six axial directions, and the label of the neighbor n determined. If the neighbor has an available bond (monomers can have one covalent bond, cross-linkers can have two), then a random number r, $0 \leq r \leq 1$, is selected. If $r \leq P_n$, P_n being the probability of a reaction occurring between the free radical and the particle type n, a reaction occurs; otherwise no reaction occurs.
   (e) If the reaction is to occur and n represents a site with no attachment to a free radical, then a bond is formed between j and n; this represents the propagation reaction. If n possesses a free radical, a bond is formed and the free radical is eliminated; this represents the termination by combination step.
   (f) If j represents an unbonded particle and j is not a void, an initiator, or an active racial, then a random neighbor n is selected. If n is a void, then the positions of n and j are switched. This allows diffusion of the constituents.
   (g) This procedure is repeated for N_{c_i} times.

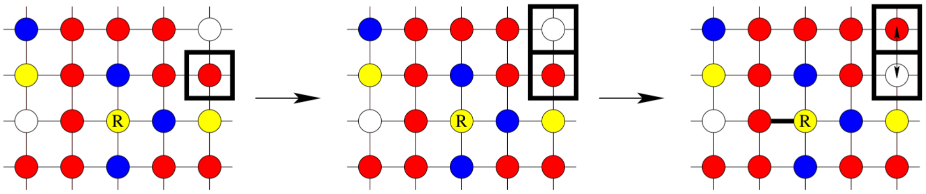
The polymerization process is illustrated schematically in Figure 2.2 and an example of a resulting configuration is given in Figure 2.3.



(a) Initialization



(b) Propagation



(c) Void Movement

Figure 2.2: A schematic of the kinetic Monte-Carlo polymerization algorithm. **Initiation** – If an initiator is randomly selected that is not a free radical, then it is made a free radical if the reaction is determined to occur. This is depicted in (a) from left to right. **Propagation** – If a free radical is randomly selected, then a random neighbor is selected. If a bond has not been formed, then a bond is formed if the reaction is determined to occur. This is depicted in (b) from left to right. **Void diffusion** – If an unreacted particle is randomly selected, then a random neighbor is selected. If that neighbor is a void, then the cell location of the void and the neighbor is switched. This is depicted in (c) from left to right.

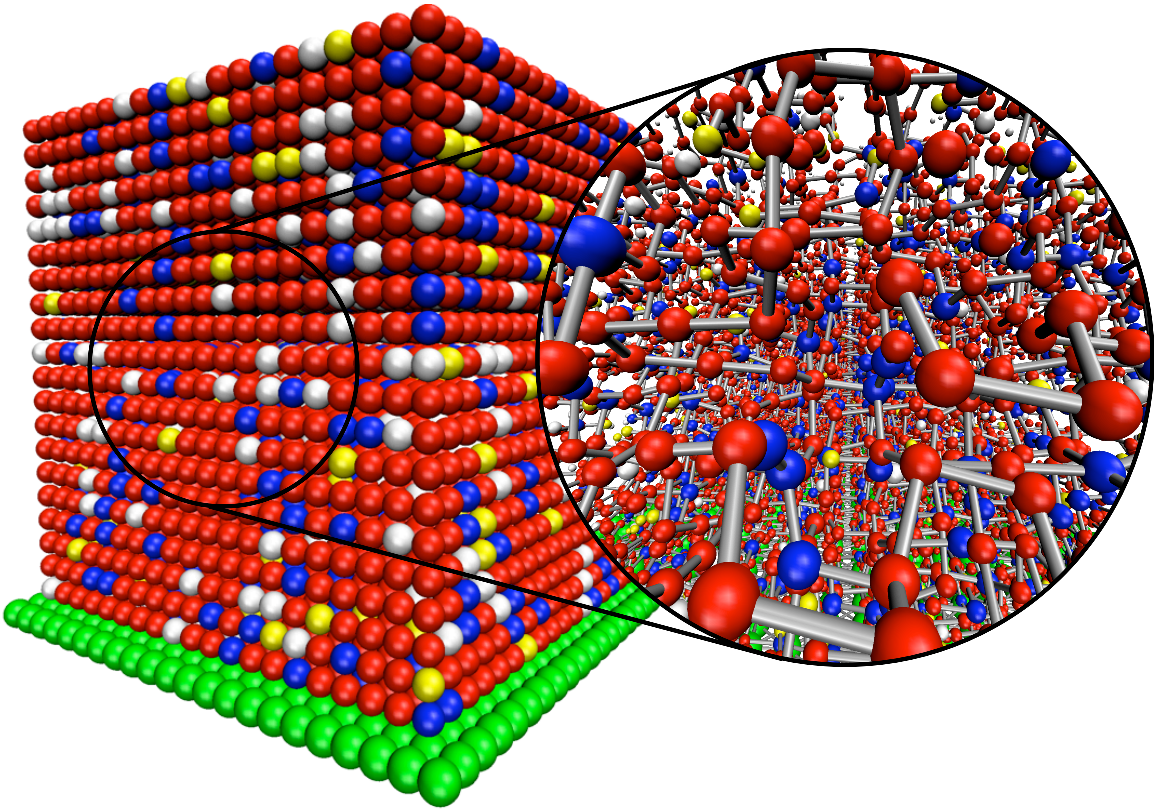


Figure 2.3: A configuration generated by the kinetic Monte-Carlo polymerization algorithm with dimensions of 21 × 101 × 21. Green spheres denote the transfer layer particles, red the monomer 1 and monomer 2, blue the cross-linkers, yellow the initiators, and white denote the voids. The zoomed portion shows the configuration of the covalent bonds formed following the relaxation of the lattice.

The polymer and its properties

At the conclusion of the kinetic Monte-Carlo process, the location of the site of each constituent and the connectivity of bonds forming the polymer chains is known. However, the motion of the molecules during the polymerization phase has been neglected. This is an inherent assumption of the mathematical model. In other words, the rate at which bonds form is assumed to be significantly faster than large, bulk movements of the chemical constituents.

It is observed experimentally that upon completion of the polymerization process, however, a volume shrinkage of approximately 20 percent occurs upon removal of the quartz template. To account for this *densification* effect, bond potentials must be assigned to the polymerized etch barrier and a mathematical model must be formulated to describe the motion of the molecules due to the formation of the bonds. To this end, it is assumed that the bulk deformation of the polymer is a quasi-static process that can effectively be modeled using molecular statics.

2.3 Mechanics of the Polymer - Molecular Statics

The goal of molecular statics is to minimize the global energy function:

$$\min_{\mathbf{v}\in U} \sum_{i=1}^{N} E_i(\mathbf{v}) \tag{2.21}$$

where N is the number of particles and $\mathbf{u}$ is the vector of displacements of all particles, i.e. $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N)$. For the remainder of the work, it is assumed that the potentials are pair potentials and that there is no external loading, i.e. $\mathbf{f}_i = 0$. This leads to a system of nonlinear equations:

$$\mathbf{g}_i = \frac{\partial E_i}{\partial \mathbf{u}_i} = \sum_{j=1}^{n_i} \frac{\partial E_{ij}}{\partial \mathbf{u}_i} = \sum_{j=1}^{n_i} \mathbf{g}_{ij} = \mathbf{0}, \qquad i = 1, \ldots, N \tag{2.22}$$

where $\mathbf{g}_i$ is the net internal force on particle i, n_i is the number of neighbor particles interacting with particle i, E_{ij} is the potential function between particles i and j and

$\mathbf{g}_{ij}$ is the force on particle i due to particle j:

$$\mathbf{g}_{ij}\left(\mathbf{u}_i, \mathbf{u}_j\right) = \frac{\partial E_{ij}\left(r_{ij}\right)}{\partial r_{ij}} \frac{\left(\mathbf{x}_j^0 + \mathbf{u}_j\right) - \left(\mathbf{x}_i^0 + \mathbf{u}_i\right)}{\left\|\left(\mathbf{x}_j^0 + \mathbf{u}_j\right) - \left(\mathbf{x}_i^0 + \mathbf{u}_i\right)\right\|} \tag{2.23}$$

where $\mathbf{x}_j^0$ is the initial position of particle j, and $r_{ij} = \|\mathbf{x}_j - \mathbf{x}_i\|$ is the distance between particles i and j. The term $\partial E_{ij}/\partial r_{ij}$ represents the magnitude of the force between particles i and j while the remaining term represents the direction of the force. Note that, even though the derivatives of the potential function may be linear, the second term induces a geometric nonlinearity. Intermolecular Hessians for both harmonic and Lennard-Jones potentials are given in Appendix C.

2.3.1 Intermolecular potentials

For the purpose of modeling the forces between the particles, each molecule is considered a point in the lattice. The covalent bonds formed during the polymerization process are modeled with harmonic spring potentials. Using the above notation,

$$E_{ij}^H = \frac{k_{ij}}{2}\left(r_{ij} - r_{ij}^0\right)^2 \tag{2.24}$$

where k_{ij} is the spring stiffness and r_{ij}^0 is the unstretched length of the spring connecting particles i and j. Weaker harmonic springs have also been used to describe van der Waal's interactions within the model. The van der Waal's potentials are only assigned to the nearest neighbors and only 18 of the 26 neighbors are included. See Fig 2.4. Tables 2.1 and 2.2 summarize the values of the spring constants used for the molecules discussed previously. The following notation is used: $(\cdot)^c$ represent parameters for covalent bonds and $(\cdot)^v$ are parameters for van der Waal's interactions. The constants for all covalent bonds are $k_{ij}^c = 10.0$ and $r_{ij}^{0c} = 3.08$.

2.4 Solution Algorithm

In this section, an algorithm to solve the minimization problem (2.21) is presented. This is taken from the optimization literature and is covered in detail in [48]. It is summarized here for the sake of completeness. The algorithms described below are implemented in the software packages PETSc [9, 8, 10] and TAO [19] and are used

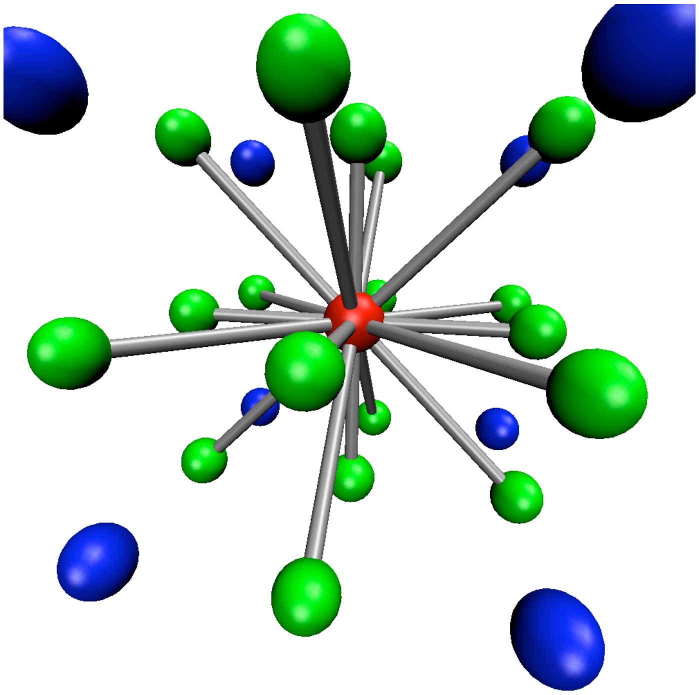


Figure 2.4: The bonding configuration for each particle. The green particles represent neighboring particles which are allowed to bond to the red; they can be covalent or van der Waal's bonds. Blue particles are not allowed to bond to the red particle.

	Monomer 1	Monomer 2	Cross-Linker	Initiator	Substrate
Monomer 1	1.1	0.8	1.2	0.9	1.0
Monomer 2	0.8	1.0	0.7	1.3	1.0
Cross-Linker	1.2	1.3	1.5	0.6	1.0
Initiator	0.9	0.7	0.5	0.6	1.0
Substrate	1.0	1.0	1.0	1.0	1.0

Table 2.1: Spring constants k_{ij}^{v} used for bonds between polymer molecules.

	Monomer 1	Monomer 2	Cross-Linker	Initiator	Substrate
Monomer 1	5.4	5.5	5.7	4.8	5.34
Monomer 2	5.5	5.2	4.9	5.9	5.34
Cross-Linker	5.7	5.9	5.3	5.8	5.34
Initiator	4.8	4.9	5.8	5.1	5.34
Substrate	5.34	5.34	5.34	5.34	5.34

Table 2.2: Spring constants r_{ij}^{0v} used for bonds between polymer molecules.

extensively in this work.

There are two approaches to solve the governing molecular statics problem: one is to solve the problem as an unconstrained optimization problem and the other is to solve the full system of nonlinear algebraic equilibrium equations. Although both appear very similar in implementation, there are very important differences that dramatically impact the robustness of the algorithm. These observations impact the multiscale algorithm described later. Both approaches are discussed below. Supporting numerical examples for the polymer model are given in the next section.

The following notation is used to describe an abstract minimization problem: the function to be minimized is $E(\mathbf{x})$, where $\mathbf{x} \in \mathbb{R}^{3N}$, the gradient of the function is $\mathbf{g} = \partial E/\partial \mathbf{x}$, and the corresponding Hessian is $\mathbf{H} = \partial \mathbf{g}/\partial \mathbf{x}$.

2.4.1 Optimization Approach

The problem is now stated in the form: find $\mathbf{x}^*$ such that

$$E(\mathbf{x}^*) = \min_{\mathbf{x} \in \mathbb{R}^{3N}} E(\mathbf{x}) \tag{2.25}$$

To construct an algorithm to solve this problem, the function $E(\mathbf{x})$ is approximated by its second-order Taylor expansion,

$$E(\mathbf{x}+\Delta\mathbf{x}) \approx E(\mathbf{x}) + \Delta\mathbf{x}^T\mathbf{g} + \frac{1}{2}\Delta\mathbf{x}^T\mathbf{H}\Delta\mathbf{x} \tag{2.26}$$

A necessary condition for a minimizer is that the gradient is zero. Thus, taking the gradient of (2.26) yields

$$\mathbf{H}\Delta\mathbf{x} = -\mathbf{g} \tag{2.27}$$

This leads to the classical Newton algorithm:

1. Let initial guess be $\mathbf{x}_0$. Set $i = 0$. Set stopping tolerance ϵ.
2. Compute $\mathbf{g}(\mathbf{x}_i)$ and $\mathbf{H}(\mathbf{x}_i)$.
3. Compute $\Delta\mathbf{x} = -\mathbf{H}^{-1}\mathbf{g}$.
4. $\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta\mathbf{x}$.
5. $i = i + 1$. If $\|\mathbf{g}(\mathbf{x}_i)\| < \epsilon$, stop. Else, goto 2.

In the present form, Newton's method can be shown to be *locally* convergent. That is, if the initial guess is "close enough" to a local minimizer, then it will converge. See [48, Theorem 3.7].

To obtain an algorithm that converges *globally* to a minimizer, in other words independent of the initial guess, the Newton method must be augmented in several ways. In particular, two features of the process must be sought: the step direction must lead to a trial solution vector which decreases the energy function sufficiently, and the Hessian must be positive definite. There are two strategies to achieve these conditions: the line search algorithm and the trust region algorithm. Both techniques are described subsequently.

Line Search

The main idea behind Newton's method with line search is that once the Newton step (2.27) is computed, the following one dimensional optimization problem is solved:

$$\min_{\alpha\in\mathbb{R}} E(\mathbf{x}+\alpha\Delta\mathbf{x}) \tag{2.28}$$

That is, one finds the *step length* in the direction of the Newton step that minimizes the energy. In practice, one solves (2.28) approximately so that the resulting correction $\mathbf{x} + \alpha\Delta\mathbf{x}$ sufficiently decreases the function. A thorough discussion is given in [48, Chapter 3] and [39, Chapter 8]. The efficient and effective determination of an appropriate α can be a challenging undertaking.

To ensure positive definiteness of the Hessian, a more substantial change to the algorithm is required. The solution of (2.27) is now solved *inexactly* using a slightly modified Conjugate Gradient (CG) algorithm. In particular, there are three important points in the CG algorithm:

1. The initial guess in the CG algorithm is the zero vector.
2. Let the current iterate in the CG algorithm be $\mathbf{x}_k$. If $k = 0$ and if $\mathbf{x}_k^T\mathbf{H}\mathbf{x}_k < 0$, perform one step of CG and terminate. If $k > 0$ and $\mathbf{x}_k^T\mathbf{H}\mathbf{x}_k < 0$, set $\mathbf{x}_k = \mathbf{x}_{k-1}$ and terminate.
3. The Newton step is the final iterate of the CG algorithm.

There are several consequences. First, if the Hessian is not positive definite, i.e. $\mathbf{x}_i^T\mathbf{H}\mathbf{x}_i < 0$, then, because the initial guess is zero, the step direction is exactly $-\mathbf{g}$, the so-called steepest descent direction. Next, since the check for positive definiteness is performed, the algorithm ensures that a step is always taken with a positive definite Hessian (actually, this is an “approximate” Hessian since the system is solved inexactly). This is true even in the case when the steepest descent step is taken, since the steepest descent step is the solution to (2.27) and employs $\mathbf{H} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Thus, the *inexact Newton method with line search* algorithm is as follows:

1. Let $\mathbf{x}_0$ be the initial guess. Set the Newton tolerance ϵ. Set $i = 0$.
2. Compute $E(\mathbf{x}_i)$, $\mathbf{g}(\mathbf{x}_i)$, and $\mathbf{H}(\mathbf{x}_i)$.
3. Solve $\mathbf{H}\Delta\mathbf{x} = -\mathbf{g}$ using CG. Terminate if:
    (a) CG tolerance reached.
    (b) $\Delta\mathbf{x}_k^T\mathbf{H}\Delta\mathbf{x}_k < 0$.
4. Solve $\min_{\alpha\in\mathbb{R}} E(\mathbf{x} + \alpha\Delta\mathbf{x})$.

5. $\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha\Delta\mathbf{x}$.

6. $i = i + 1$. If $\|\mathbf{g}(\mathbf{x}_i)\| < \epsilon$, stop. Else, goto 2.

Trust Region

The main idea behind the trust region algorithm is, to some degree, the converse of the line search method. Instead of first choosing the step direction and then the step length along that direction, the step length is first chosen, the so-called *trust region*, and then the step direction is chosen so that the function is minimized inside the trust region. This is stated mathematically as the following constrained optimization problem. Let

$$m_i(\Delta\mathbf{x}) = E(\mathbf{x}_i) + \Delta\mathbf{x}^T\mathbf{g}(\mathbf{x_i}) + \frac{1}{2}\Delta\mathbf{x}^T\mathbf{H}(\mathbf{x_i})\Delta\mathbf{x} \tag{2.29}$$

where i represents the current Newton iteration. Then, the trust region problem is

$$\min_{\mathbf{x}\in\mathbb{R}^{3N}, \|\mathbf{x}\|\leq\Delta_i} m_i(\mathbf{x}) \tag{2.30}$$

where Δ_i is the trust region radius. Note that if $\mathbf{H}$ is positive definite and $\|\mathbf{H}^{-1}\mathbf{g}\| \leq \Delta_i$, then the solution to the problem is the Newton step. Thus, one can see that a sequence of problems must be solved to find step directions that solve the minimization problem. The exact solution of this problem is cost-prohibitive, but efficient and robust approximations can be constructed. The method is actually quite similar to the inexact Newton line search method.

The inexact Newton trust region, or Steihaug algorithm [68], is identical to the line search with a few small differences.

1. Let $\mathbf{x}_0$ be the initial guess. Set the Newton tolerance ϵ. Set $i = 0$.

2. Compute $E(\mathbf{x}_i)$, $\mathbf{g}(\mathbf{x}_i)$, and $\mathbf{H}(\mathbf{x}_i)$.

3. Solve $\mathbf{H}\Delta\mathbf{x} = -\mathbf{g}$ using CG. Terminate if:

    (a) CG tolerance reached.

    (b) $\Delta\mathbf{x}_k^T\mathbf{H}\Delta\mathbf{x}_k < 0$.

    (c) $\|\Delta\mathbf{x}_k\| \geq \Delta_i$.

4. Adjust Δ_i. Goto 3, if necessary.

5. $\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta\mathbf{x}$.
6. $i = i + 1$. If $\|\mathbf{g}(\mathbf{x}_i)\| < \epsilon$, stop. Else, goto 2.

An additional check has been added to the CG algorithm to ensure that the current solution iterate is still within the trust region. Thus, the algorithm ensures a descent direction and a solution that is within the trust region. To ensure that the algorithm maintains asymptotic quadratic convergence, the trust region must be adjusted at each Newton step so that, when the Newton iterate is near a solution, the trust region radius does not constrain the Newton step.

Define

$$\rho_i = \frac{E(\mathbf{x}_i) - E(\mathbf{x}_i + \Delta\mathbf{x})}{m_i(0) - m_i(\Delta\mathbf{x})} \tag{2.31}$$

The number ρ_i measures how accurate the Newton (quadratic) approximation is compared to the actual function. In other words, it gives an indication of how "trust-worthy" the approximation is. Then, the trust region is adjusted based on ρ_i: if ρ_i is small (or negative), the approximation is determined to be poor and the trust region is reduced; if ρ_i is close to one, then the approximation is good (the model is trust-worthy) and the trust region size is increased. An example of an algorithm is as follows:

1. Given the CG solution $\Delta\mathbf{x}$, compute ρ_i.
2. If $\rho_i < \frac{1}{4}$, then $\Delta_i = \frac{1}{4}\|\Delta\mathbf{x}\|$. $\mathbf{x}_{i+1} = \mathbf{x}_i$.
3. Else, if $\rho_i > \frac{3}{4}$ and $\|\Delta\mathbf{x}\| = \Delta_i$, then $\Delta_i = 2\Delta_i$. $\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta\mathbf{x}$.

2.4.2 Nonlinear Equations Approach

It is now assumed that no energy function is present that is to be minimized, but rather the goal to find the zeroes of a vector valued nonlinear function $\mathbf{r}(\mathbf{x}) \in \mathbb{R}^{3N}$: find $\mathbf{x}^*$ such that

$$\mathbf{r}(\mathbf{x}^*) = 0 \tag{2.32}$$

Note that, if the function $\mathbf{r}(\mathbf{x})$ derives from an energy function, $\mathbf{r}(\mathbf{x}) = \mathbf{g}(\mathbf{x})$. The purpose here is illustrate the differences between the optimization and nonlinear equation algorithms and the effects on convergence properties. Let $\mathbf{J}(\mathbf{x})$ represent the Jacobian of the function $\mathbf{r}(\mathbf{x})$. Then, the classical Newton algorithm is

1. Let initial guess be $\mathbf{x}_0$. Set $i = 0$. Set stopping tolerance ϵ.
2. Compute $\mathbf{r}(\mathbf{x}_i)$ and $\mathbf{J}(\mathbf{x}_i)$.
3. Compute $\Delta\mathbf{x} = -\mathbf{J}^{-1}\mathbf{r}$.
4. $\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta\mathbf{x}$.
5. $i = i + 1$. If $\|\mathbf{r}(\mathbf{x}_i)\| < \epsilon$, stop. Else, goto 2.

As before, this algorithm is locally convergent and must be enriched to obtain global convergence. This can be accomplished by augmenting the algorithm with line search and trust region algorithms as before, but the challenge here is that there is no function to decrease. Therefore, a merit function is introduced to serve this purpose. A popular choice, related to least squares algorithms, is to define

$$f(\mathbf{x}) : \mathbb{R}^{3N} \to \mathbb{R}, f(\mathbf{x}) = \|\mathbf{r}(\mathbf{x})\|^2 \tag{2.33}$$

Then, the line search and trust region algorithms may be utilized using this function. The issue is that, while some minima of this function correspond to roots of $\mathbf{r}(\mathbf{x})$, other minima also exist that can be attractors for solutions and, therefore, global convergence properties are not as robust as the optimization algorithms. Furthermore, there is no notion of “negative curvature”. Since there is no function to minimize, the check for negative curvature does not exist in these algorithms. This can significantly decrease the robustness of the algorithm, as seen in numerical examples.

2.5 Numerical Results

In this section, several results are shown illustrating both the character of solutions of the base model and the convergence properties of the various algorithms discussed in the previous section. Both uniform lattices and polymer lattices are shown. In all cases below, the relative residual decrease required for convergence of the nonlinear solver is a tolerance of 10^{-8}. All other parameters are the default parameters in the PETSc and TAO packages.

A uniform lattice $10 \times 10 \times 10$ is considered with parameters $k_{ij} = 3.0$, axial equilibrium lengths $r_{ij}^{0a} = 3.0$, and diagonal equilibrium lengths $r_{ij}^{0d} = 3.0\sqrt{2}$; the initial lattice spacing is 4.0 units. This lattice is equilibrated using the standard Newton

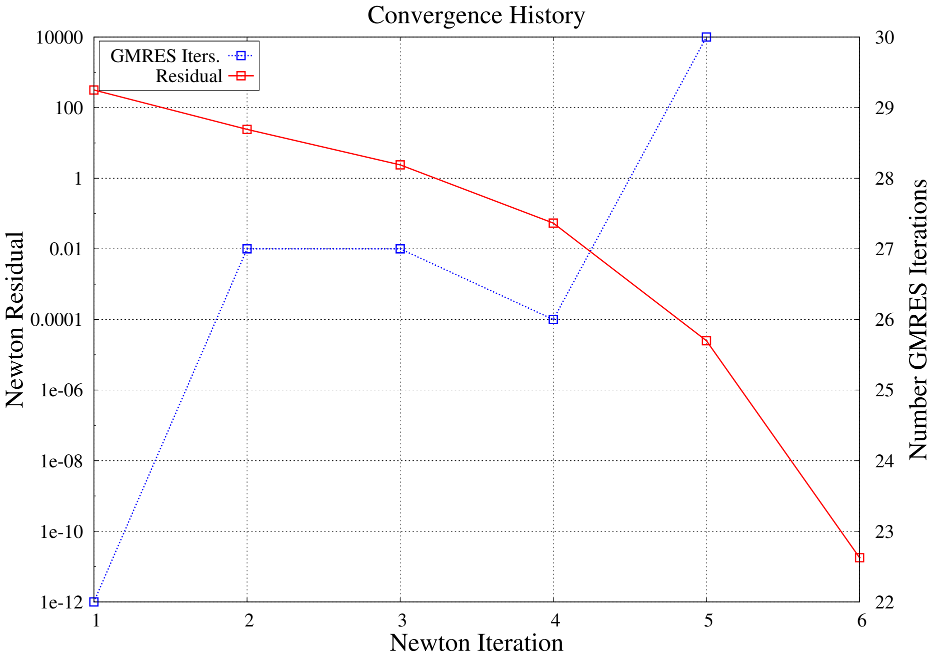


Figure 2.5: Convergence behavior of a uniform $10 \times 10 \times 10$ lattice using the standard Newton method. The red curve shows the value of the residual (gradient) and the blue curve is the number of GMRES iterations required at each step.

method. Figure 2.5 illustrates the convergence behavior. A $10 \times 10 \times 10$ polymer lattice is considered using the same solution methodology. The spring parameter are listed in Tables **??**-2.2 and the initial lattice spacing is 6.0 units. Figure 2.6 shows the convergence results. As can be seen, the uniform lattice converges uniformly to a minimum in six steps and achieves asymptotic quadratic convergence. On the other hand, the polymer lattice fails to converge: the residual oscillates and never converges, illustrating the need for more robust techniques for polymer lattices.

Augmented Newton methods using the nonlinear equations approach are now used to attempt equilibrating the lattice. Figure 2.7 shows the results of both the line search and trust-region augmentations. In the case of line search, the system becomes too ill conditioned to solve effectively, while the trust-region parameter becomes too small in the case of the trust-region augmentation. Note that if the GMRES solver is replaced by a direct solver in the line search case, the method still fails to converge, but now the line search fails: $\alpha < 10^{-12}$.

Now the same polymer $10 \times 10 \times 10$ is solved using the optimization approach. Figure 2.8 shows results from both the line search and trust-region methodologies.

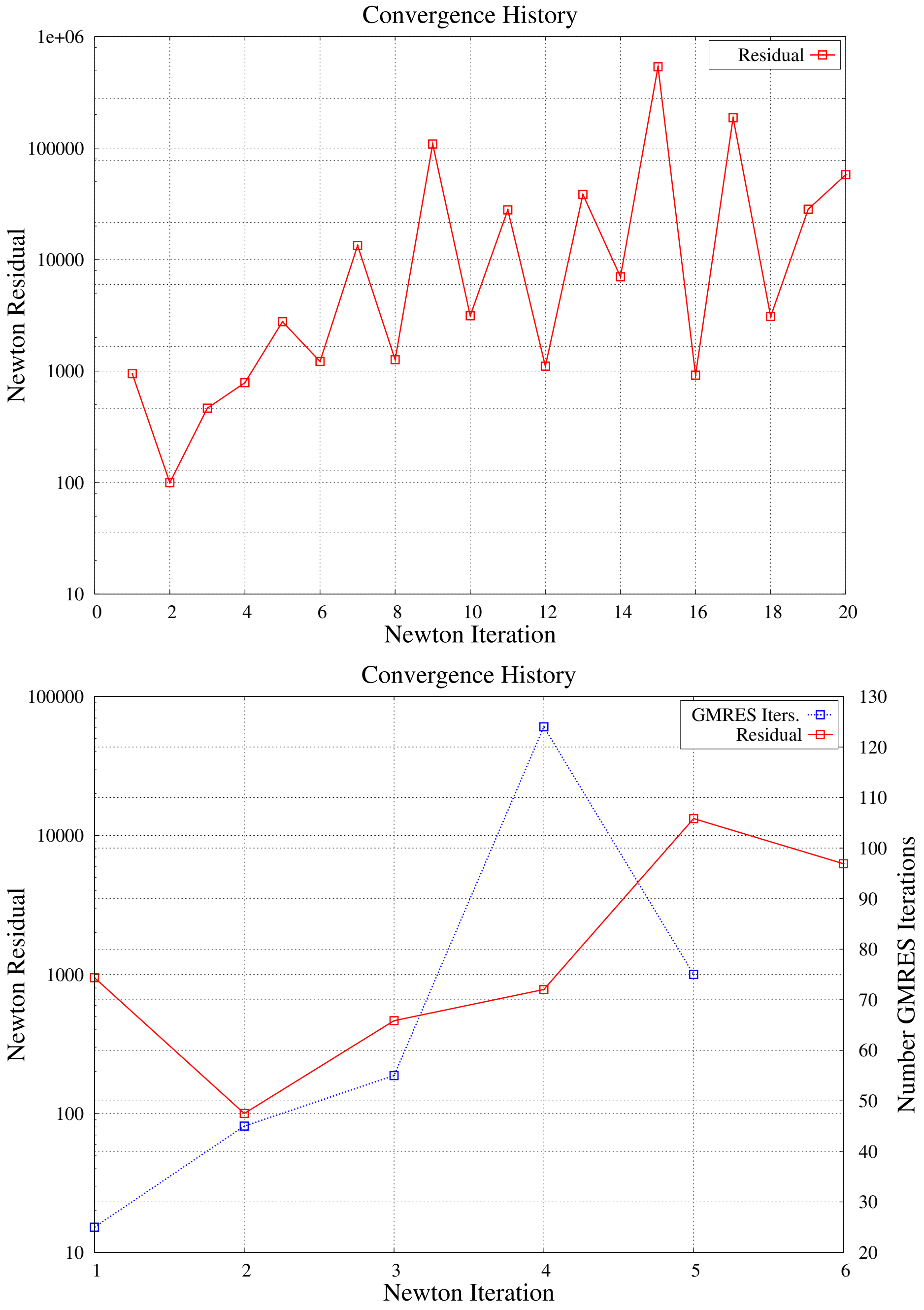
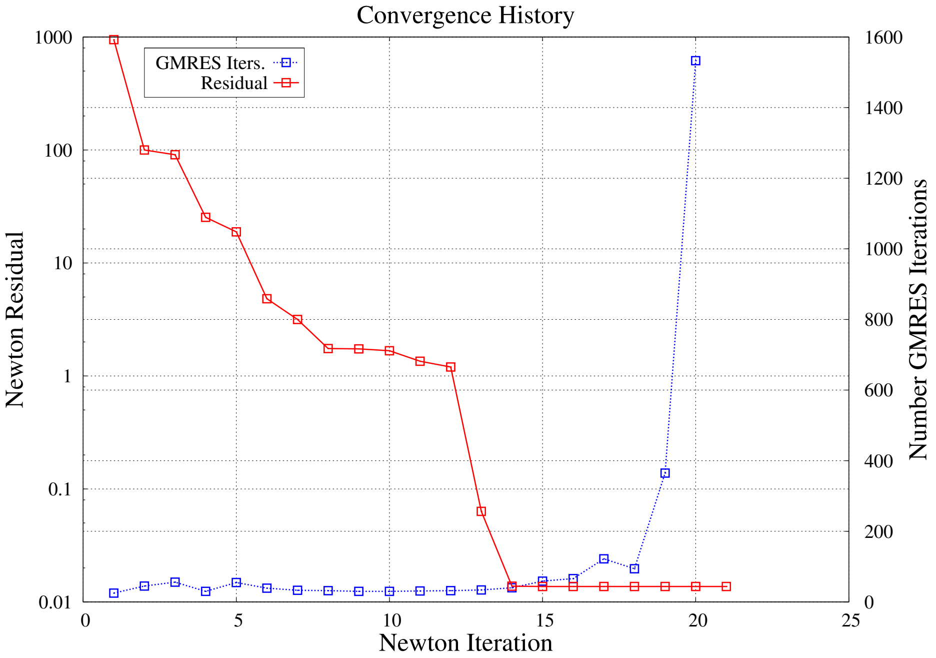


Figure 2.6: Convergence behavior of a 10 × 10 × 10 polymer lattice using the standard Newton method. The red curve shows the value of the residual (gradient) and the blue curve is the number of GMRES iterations required at each step. The top set of curves was generated using the solver Super LU [41] to solve the linear system for the newton step (hence no blue curve). The procedure was manually terminated after 20 iterations. The bottom set of curves used GMRES with ILU preconditioning. The solution process was terminated because GMRES failed to converge for iteration six.



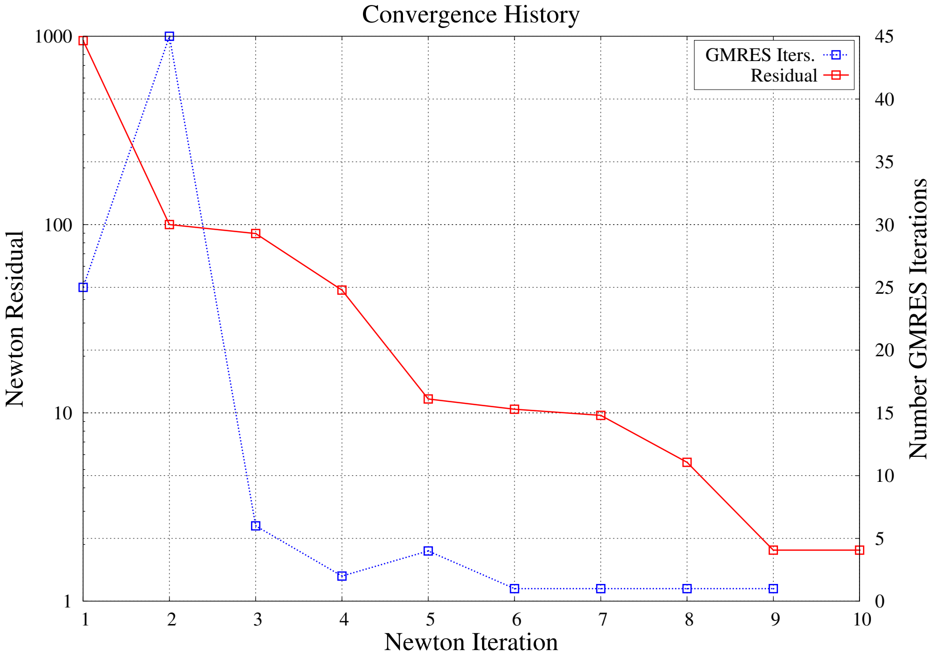


Figure 2.7: Convergence behavior of a 10 × 10 × 10 polymer lattice using enriched Newton methods in the nonlinear equations approach. The red curve shows the value of the residual (gradient) and the blue curve is the number of GMRES iterations required at each step. The top set of curves was generated using Newton with line search. The procedure terminated because GMRES failed to converge at the final step. The bottom set of curves was computed using Newton with trust-region. The algorithm terminated due to the trust-region size $\Delta < 10^{-13}$. Note that the trust-region size restricted in the number of GMRES iterations following iteration 2.

Note that in the work done in resizing the trust-region is not plotted, i.e. the iterations where the trust-region was shrank and the Newton step recalculated are not shown; although very little extra work is done, it neglects to show where negative curvature was first encountered since the step may have terminated due to trust-region restrictions instead. Figure 2.9 shows the configurations for several Newton steps. Based on these images, it would seem that the first few Newton iterations capture the bulk deformation of the material and the ensuing steps adjust the small deformations of individual particles. Furthermore, based on the convergence behavior, it seems the challenge with this problem is that the minima may be very closely spaced requiring many Newton iterations to find a minimum.

Consider now a $30 \times 30 \times 30$ polymer lattice using the same parameters as before. Convergence curves are shown in Figure 2.10. Both solutions converge, but the trust-region does so in a more uniform way. The line search approach is hindered by indefinite pre-conditioners at several Newton iterations that hinder the uniformity of the convergence. Figure 2.11 shows the resulting equilibrium configuration. The convergence behavior is similar to the $10 \times 10 \times 10$ case, but there are many more Newton iterations indicating that perhaps the function becomes less smooth as the dimension is increased, i.e. more particles gives many more possible minima. This notion is also supported by the $110 \times 110 \times 110$ lattice results shown in Figures 2.13 and 2.12. This problem consists of over 4.1 million unknowns and was solved on 64 processors taking a total 65 CPU hours to solve. Indeed, over 3,600 Newton iterations were required to attain equilibrium.

Results have been shown illustrating the equilibrium configuration of the base model as well as the convergence behavior of the nonlinear solvers used. Based on these results, the inexact Newton trust-region method seems to be the most efficient and robust algorithm for the base model. Although the nonlinear equations implementation is very similar, the inability to detect negative curvature is a major hindrance to obtain convergence. Furthermore, the problem becomes much more challenging as the problem size increases as more Newton iterations are required.

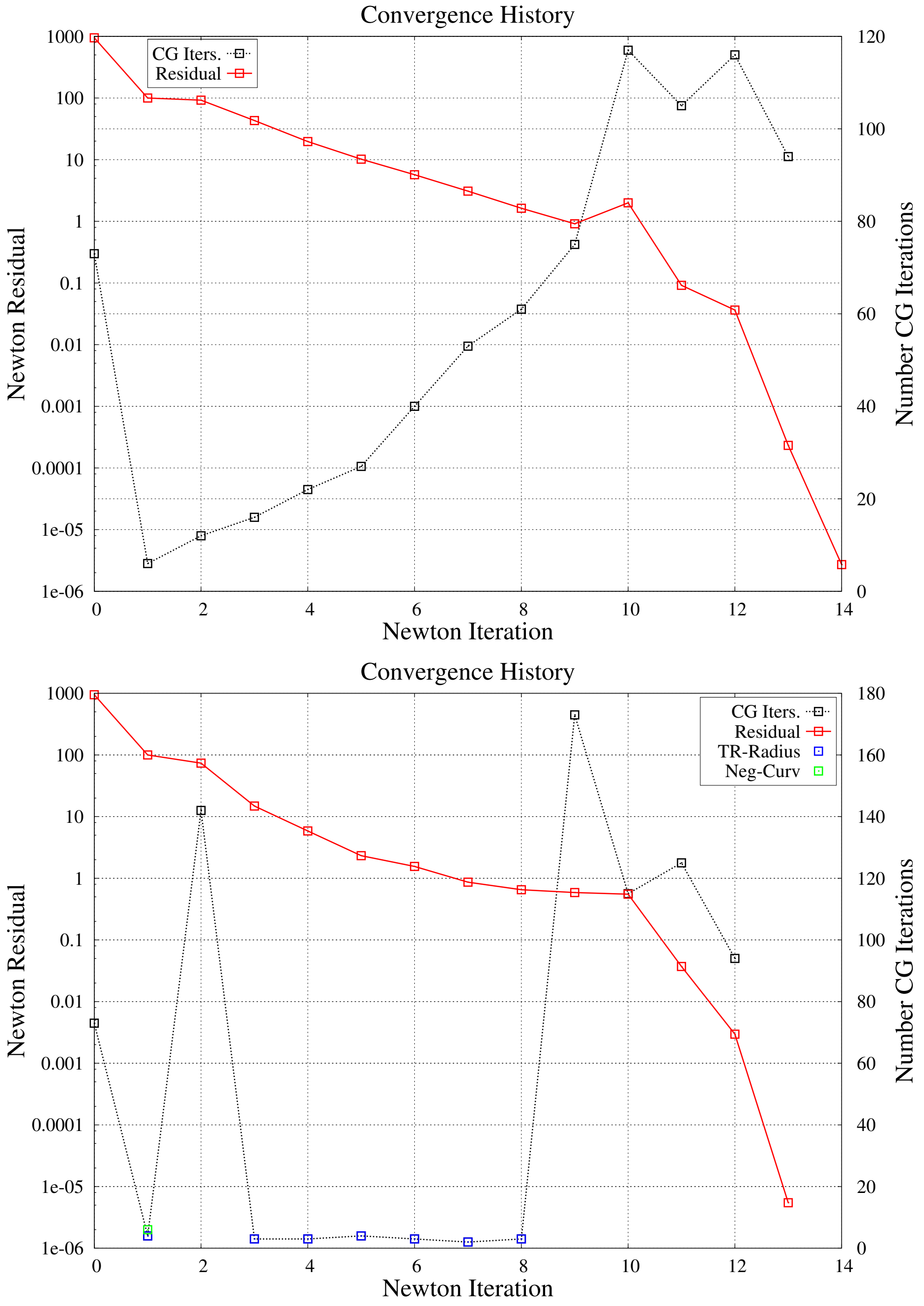


Figure 2.8: Convergence behavior of a $10 \times 10 \times 10$ polymer lattice using unconstrained optimization algorithms. The red curve shows the value of the residual (gradient) and the black curve is the number of C-G iterations required at each step. The top set of curves was generated using Newton with line search. The bottom set of curves was computed using Newton with trust-region. Note that the blue boxes show where the trust-region size restricted the size of the Newton step while black boxes mean the linear solver converged fully. In both cases, the algorithms terminated successfully: the relative tolerance of the residual was reduced by the specified amount.

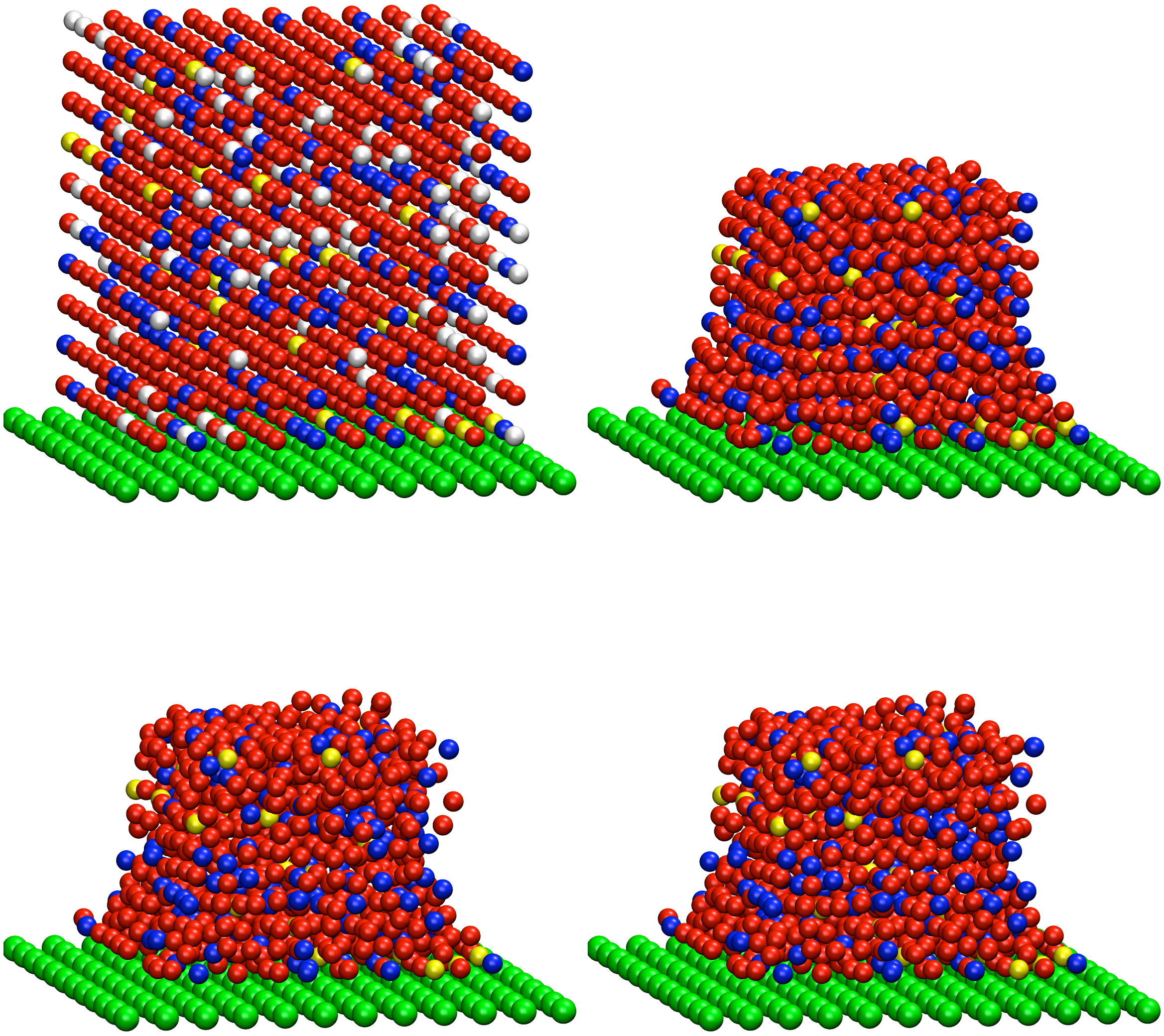


Figure 2.9: Deformations of $10 \times 10 \times 10$ polymer lattice using the inexact Newton with trust-region algorithm. Newton steps 0, 1, 5, and 13 are shown. Red particles denote the monomers, blue the crosslinkers, yellow the initiators, green the (fixed) transfer layer, and white are the voids (only shown at step 0). All images of the base model deformations are rendered using the freely available package Visual Molecular Dynamics (VMD). [38]

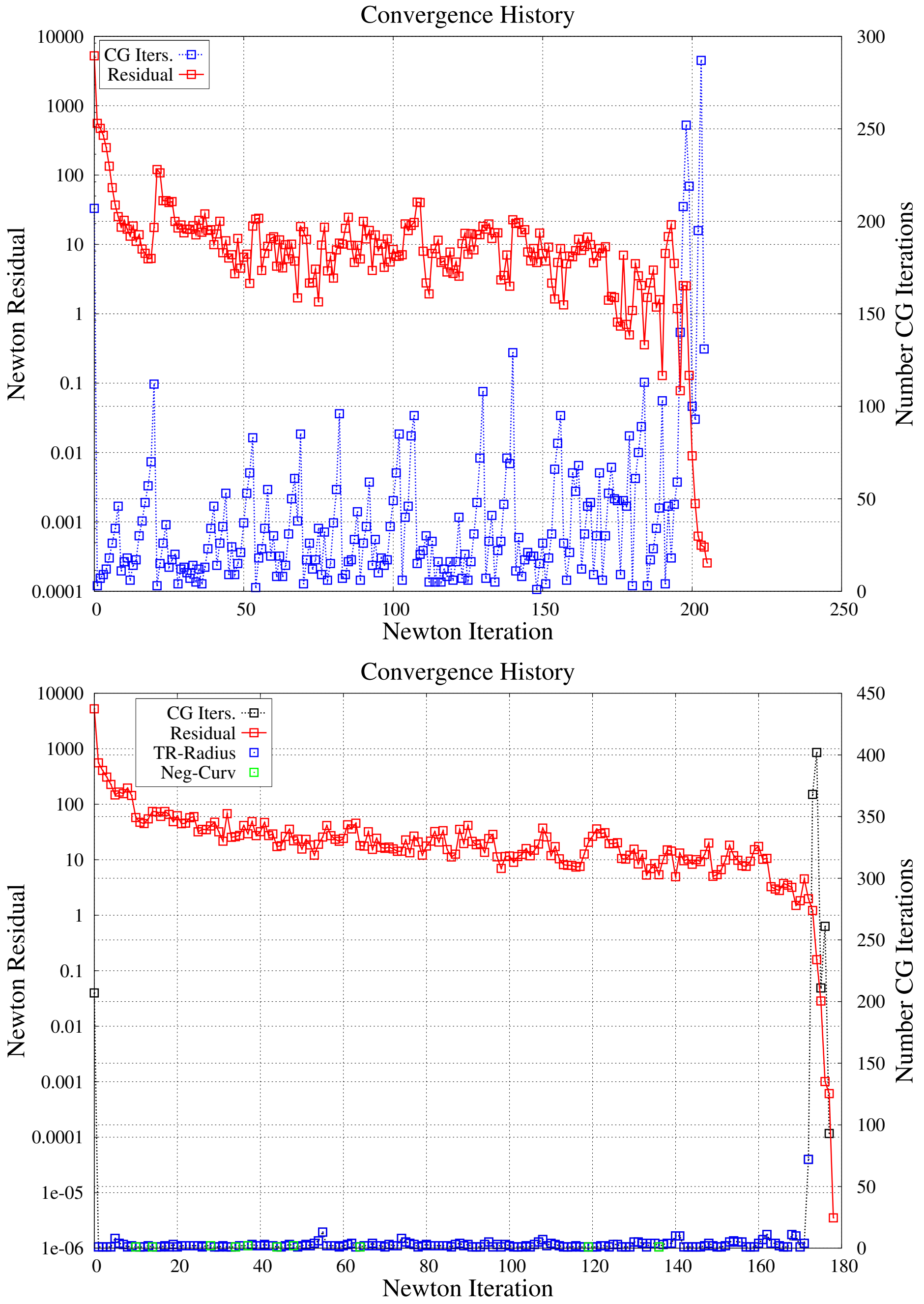


Figure 2.10: Convergence behavior of a 30 × 30 × 30 polymer lattice using unconstrained optimization algorithms. The red curve shows the value of the residual (gradient) and the black curve is the number of C-G iterations required at each step. The top set of curves was generated using Newton with line search. The bottom set of curves was computed using Newton with trust-region. Note that the blue boxes show where the trust-region size restricted the size of the Newton step and green boxes show that negative curvature was encountered. In both cases, the algorithms terminated successfully: the relative tolerance was reduced by the specified amount.

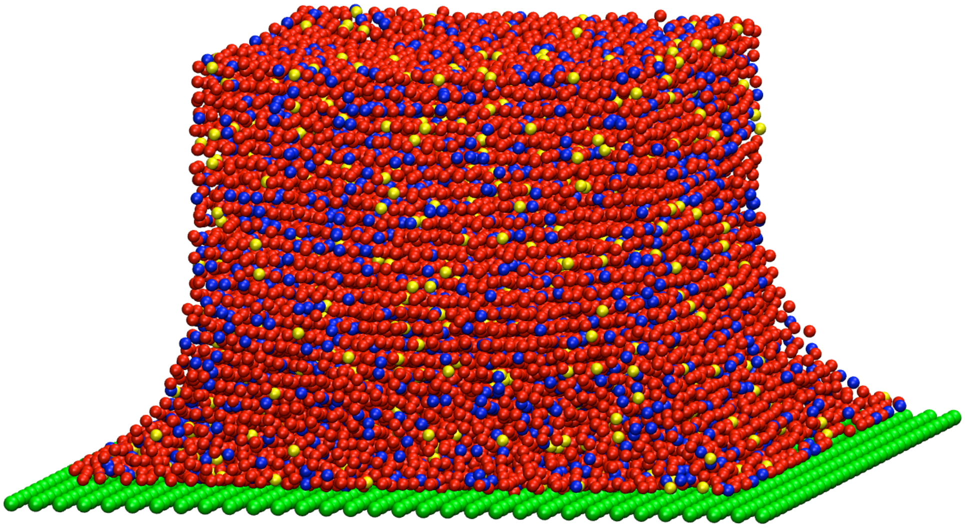


Figure 2.11: Equilibrium configuration of a 30 × 30 × 30 polymer lattice using the inexact Newton with trust region algorithm. Red particles denote the monomers, blue the crosslinkers, yellow the initiators, and green the (fixed) transfer layer.

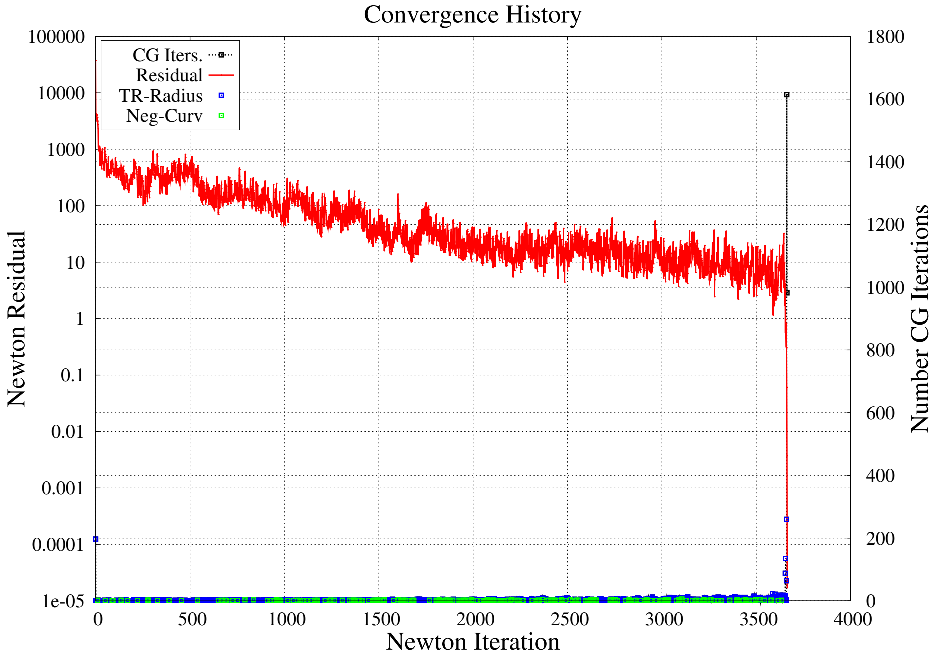


Figure 2.12: The figure shows the convergence behavior for equilibrating a 110 × 110 × 110 polymer lattice using the inexact Newton trust-region algorithm. Note that the blue boxes show where the trust-region size restricted the size of the Newton step and green boxes depict where negative curvature was detected, while black boxes mean the linear solver converged fully.

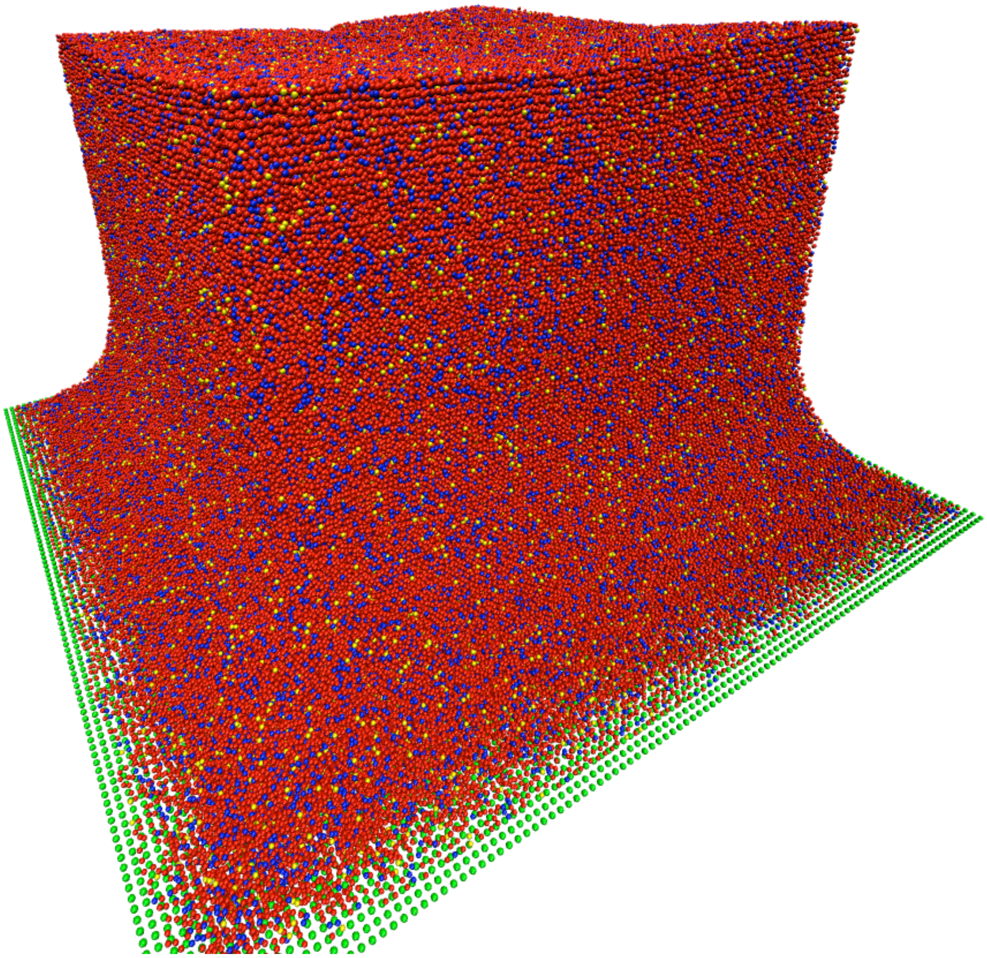


Figure 2.13: The figure depicts an equilibrium configuration of a $110 \times 110 \times 110$ polymer lattice using the inexact Newton with trust-region algorithm. Red particles denote the monomers, blue the crosslinkers, yellow the initiators, and green the (fixed) transfer layer.

Chapter 3

Development of a Continuum Model

3.1 Introduction

A critical step in the multiscale modeling procedure developed here is the construction of surrogate models, as in (1.10), that represent events seen at larger scales than the base model, but are close to the base model in some sense. Generally, coarser-scale surrogates may result from averaging the features of fine-scale models through various homogenization methods or ensemble-averaging techniques. In this case, a continuum model is chosen, but whose corresponding constitutive equation coefficients are unknown. A scheme is described below that is designed to determine these unknown coefficients using "virtual" (numerical) experiments that are performed on a representative volume element (RVE) of polymer material. Once the continuum model is defined completely, then finite element approximations can be used to determine the solution to the governing partial differential equations (PDE).

3.1.1 Virtual Experiments on RVE's

The polymerization process described earlier does not involve any features that would lead to macroscale inhomogeneities or anisotropies, and the monomer and crosslinker constituents were designed to avoid or minimize rate effects. Thus, the assumption that the process leads to a material behaving as an amorphous elastomer seems reasonable. The macroscale model of the polymerized etch barrier is then that of a homogeneous, isotropic, hyperelastic material with a stored energy per unit volume W relative to a reference configuration $\Omega_0 \subset \mathbb{R}^3$.

Let $\mathbf{X}$ denote the labels of material particles of the body, with X_i the cartesian coordinates of particle positions in the reference configuration. The motion $\varphi : \overline{\Omega}_0 \to \overline{\Omega}$ is a differentiable, orientation preserving map that takes particles into positions $\mathbf{x} = \varphi(\mathbf{X})$ in the current ("deformed") configuration, and $\mathbf{F}(\mathbf{X}) = \nabla_{\mathbf{X}}\varphi$ is the deformation gradient tensor. The right Cauchy-Green deformation tensor is denoted $\mathbf{C} = \mathbf{F}^T\mathbf{F}$. A fundamental condition on the stored energy function W is that it be form-invariant under all changes in the observer frame of reference, which, generally, means that W must depend on invariants of the deformation. One way of guaranteeing this invariance is to write W as a function of the principal invariants of $\mathbf{C}$:

$$W = W(I_1, I_2, I_3) \tag{3.1}$$

where

$$\begin{aligned} I_1 &= tr\mathbf{C} = \lambda_1 + \lambda_2 + \lambda_3 \\ I_2 &= \frac{1}{2}\left(tr\mathbf{C}\right)^2 - \frac{1}{2}tr\left(\mathbf{C}\right)^2 = \lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3 \\ I_3 &= \det\mathbf{C} = \lambda_1\lambda_2\lambda_3 \end{aligned} \tag{3.2}$$

and λ_i are the principal stretches of material line elements.

The virtual testing procedure involves the following steps:

1. A model of a cube $D \subset \mathbb{R}^3$ of polymerized material, initially of size $L_0 \times L_0 \times L_0$ with $N_x \times N_y \times N_z = N_D$ particles, is generated using the kinetic Monte-Carlo algorithm described in Section 2.2.7, and is allowed to assume a relaxed (densified) equilibrium configuration after constraints sufficient to eliminate rigid translations and rotations are applied. The cube D is the initial RVE.

2. Uniform tractions are applied over opposite faces of the cube by assigning values of the net force on molecules residing on the near-planar boundaries of D.

3. The resulting principal stretches $\lambda_1, \lambda_2, \lambda_3$ are calculated by computing the length of the stretched RVE and taking the ratio with the initial length of the relaxed RVE: $\lambda_1 = L^x/L_0^x$, etc. These lengths are calculated by averag-

ing the positions of the particles on each face and computing the difference of these averages. Furthermore, the Jacobian $J = \det \mathbf{F} = \sqrt{\lambda_1\lambda_2\lambda_3}$ is measured (calculated) from the RVE.

4. The total lattice energy E_D in D is computed in the relaxed and stretched configurations:

   $$E_D = \frac{1}{2}\sum_{i=1}^{N_D}\sum_{j=1}^{N_D} E_{ij}(\mathbf{u})$$

   N_D being the number of unconstrained lattice points in D.

5. The total energy density in D is equated to the continuum energy density in D:

   $$\frac{E_D}{V_0} = W \tag{3.3}$$

   Furthermore, for the cases of uniaxial and biaxial extension, exact solutions can be formulated for the continuum model and the measured stretches λ_i can be compared to the expected values.

This process is depicted in Figure 3.1.

By substituting various assumed forms of W into (3.3), for independent virtual tests, the constitutive coefficients can be determined. Some important aspects of this process are noted:

1. The constants determined by this process should be independent of the size of the RVE; thus the dimension L_0, i.e. $N_x \times N_y \times N_z$, of the domain D should be increased until the computed values of the material parameters do not change.

2. A single molecular model of the RVE represents only one realization of the polymeric structure. Thus, a large number of such realizations should be generated to determine the statistical variations of the material parameters with sufficient accuracy.

3. The form of W assumed in the process should represent a stable characterization of the material, relatively insensitive to small perturbations in the molecular model.

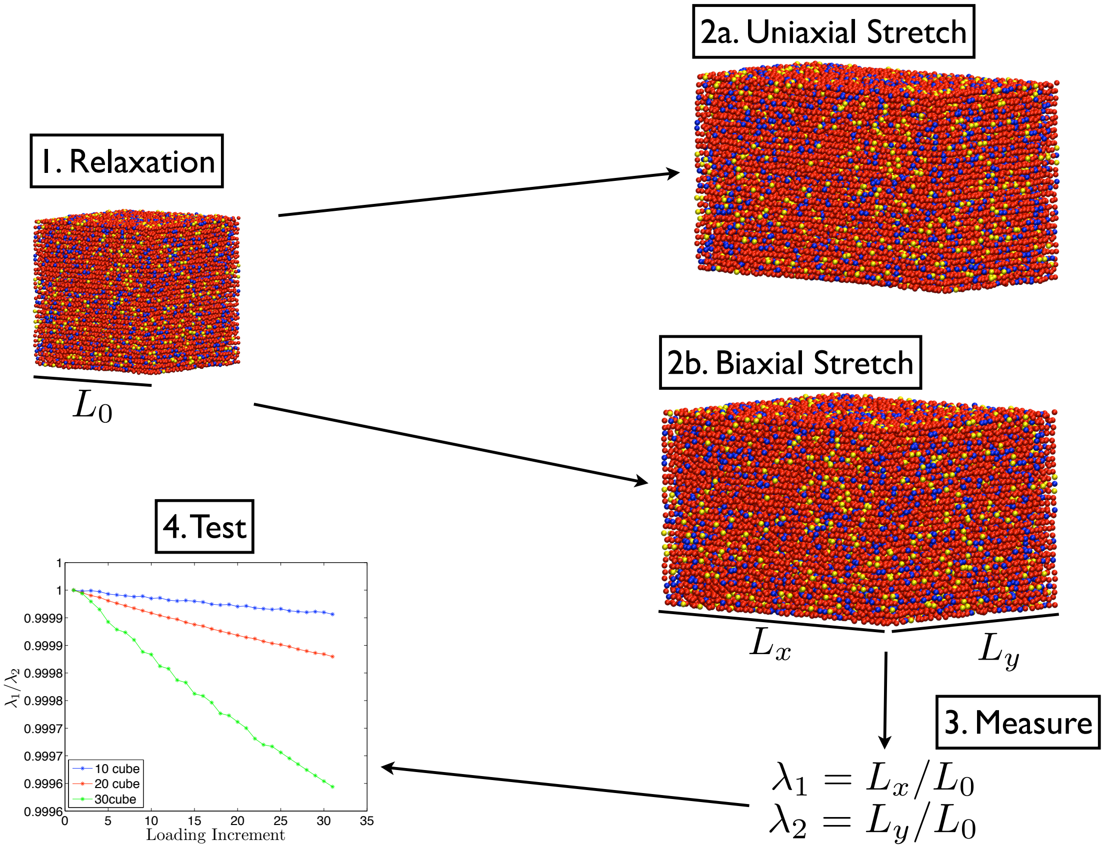


Figure 3.1: RVE testing procedure. Step 1. Relaxation of the polymer lattice. Step 2. Deformation (uniaxial and biaxial stretches). Step 3. Measurements of $V, \lambda_1, \lambda_2, \lambda_3, E$. Step 4. Test for symmetry and other necessary conditions to ensure the RVE is large enough.

3.2 Constitutive Equations

One possible generic form of W is the power series [56]

$$W = \sum_{r=1}^{M} \sum_{s=1}^{M} \sum_{t=1}^{M} C_{rst}(I_1 - 3)^r (I_2 - 3)^s (I_3 - 1)^t \tag{3.4}$$

(where $W = 0$ when $\mathbf{C} = \mathbf{I}$). Traditionally, one can perform laboratory experiments that are designed to determine the coefficients C_{rst} and the number of terms M for prescribed values of I_1, I_2, and I_3. Instead, in this work, *virtual* experiments are performed using RVE's of the polymer model. The following forms are considered:

$$\begin{aligned} W &= C_1(I_1 - 3 - \ln\sqrt{I_3}) + C_2(\sqrt{I_3} - 1)^2 \\ W &= C_1(I_1 - 3) + C_2(I_2 - 3) + C_3(\sqrt{I_3} - 1)^2 - (2C_1 + 4C_2)\ln\sqrt{I_3} \end{aligned} \tag{3.5}$$

These equations represent the compressible versions of the classical Neo-Hookean and Mooney-Rivlin materials [24]. Actually, the development of these constitutive equations is inspired by early work in the statistical mechanics of polymer networks and to arguments related to macroscale experiments on elastomers [61, 72]. This foundation in polymer modeling is the motivation for selecting these constitutive equations.

3.3 Numerical RVE Experiments

3.3.1 Determination of RVE Size

Here, the details of the numerical RVE experiments are discussed. Following the previous discussion, the size of the RVE must be determined so that the constants to be computed for the continuum model are stable with respect to the RVE size. To determine the RVE size, an RVE is subjected to uniaxial extension with lateral contraction and biaxial extension. Since the material behaves as an isotropic homogenous material on the scale of the RVE, then it is expected that certain symmetries will be present. In the case of uniaxial extension in the x-direction, $\lambda_2 = \lambda_3$. In biaxial extension in the x-direction and y-direction, $\lambda_1 = \lambda_2$. Furthermore, in both cases the ratio of the volume change $V/V_0 = \sqrt{I_3}$, and it is expected the energy density should converge as the RVE size is increased. Thus, in the RVE experiment, $\lambda_1, \lambda_2, \lambda_3$

are computed by taking the ratio of the deformed length in each direction with the original length, and the volume of both the deformed and original configurations are calculated. This procedure was executed on polymer RVE's of size $10 \times 10 \times 10$, $20 \times 20 \times 20$, and $30 \times 30 \times 30$. Figure 3.2 shows the energy density and the volume comparison for uniaxial loading tests while Figure 3.3 shows the symmetry tests for uniaxial and biaxial loadings. As can be seen, as the RVE size is increased to $30 \times 30 \times 30$, the properties converge as expected to quite tight tolerances. Thus, a $30 \times 30 \times 30$ size RVE will be used to determine the continuum coefficients.

3.3.2 Parameter Fit from RVE Data

With the RVE size determined and data collected from deformations on the RVE, a least squares fitting procedure is used to fit parameters for the continuum constitutive equation. There are three sets of data used to fit the parameters: the energy density for both uniaxial and biaxial deformations, the relationship between λ_2 and λ_3 in the uniaxial deformation, and the relationship between λ_1 and λ_2 in the biaxial deformation for the different loading values. In the first case, this merely means fitting $W(I_1, I_2, I_3) = E/V_0$ where I_1, I_2, I_3 are computed from the measured stretches. In the latter two cases, these relationships come from the exact solution to the uniaxial and biaxial deformations; the details of the derivation and the results are given in Appendix D.

For this fitting procedure, the body is stretched 30% in 20 incremental steps for the uniaxial and biaxial deformations. Once the data has been collected, the function *lsqcurvefit* in the program MATLAB is used to fit the data. The resulting parameters are tabulated in Table 3.1 and the corresponding curve fits are shown in Figure 3.4 and Figure 3.5.

Table 3.1: Results of parameter fit for the Compressible Neo-Hookean and the Compressible Mooney-Rivlin materials based upon the RVE experiments on a $30 \times 30 \times 30$ polymer lattice.

Model	C_1	C_2	C_3
Comp. Neo-Hookean	0.72	0.57	N/A
Comp. Mooney-Rivlin	0.67	0.23	0.22

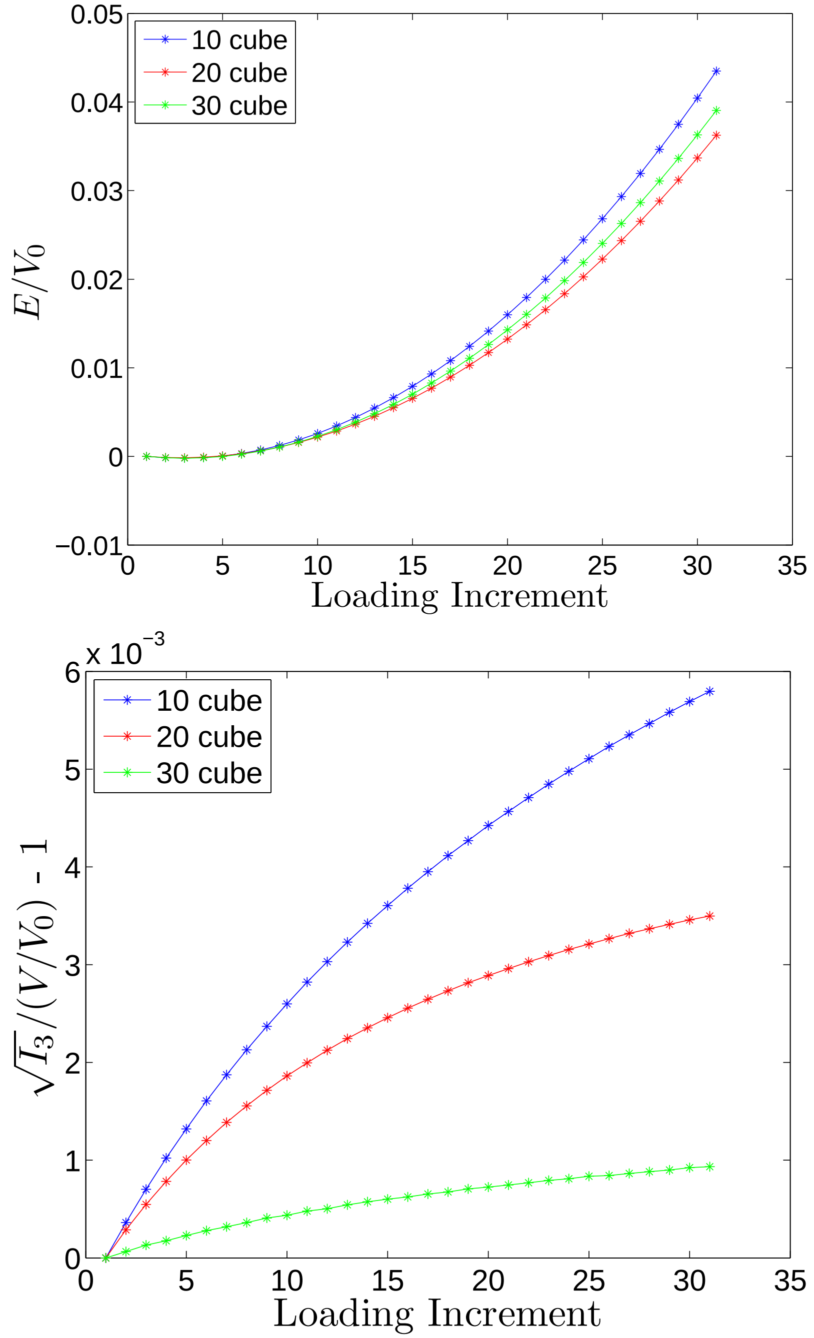


Figure 3.2: RVE's of the polymer are deformed under uniaxial loading. The top curve illustrates the energy density for the 10, 20, and 30 cubes while the bottom curve shows the ratio $\sqrt{I_3}/(V/V_0)$; if the polymer was perfectly symmetric, this ratio would be one.

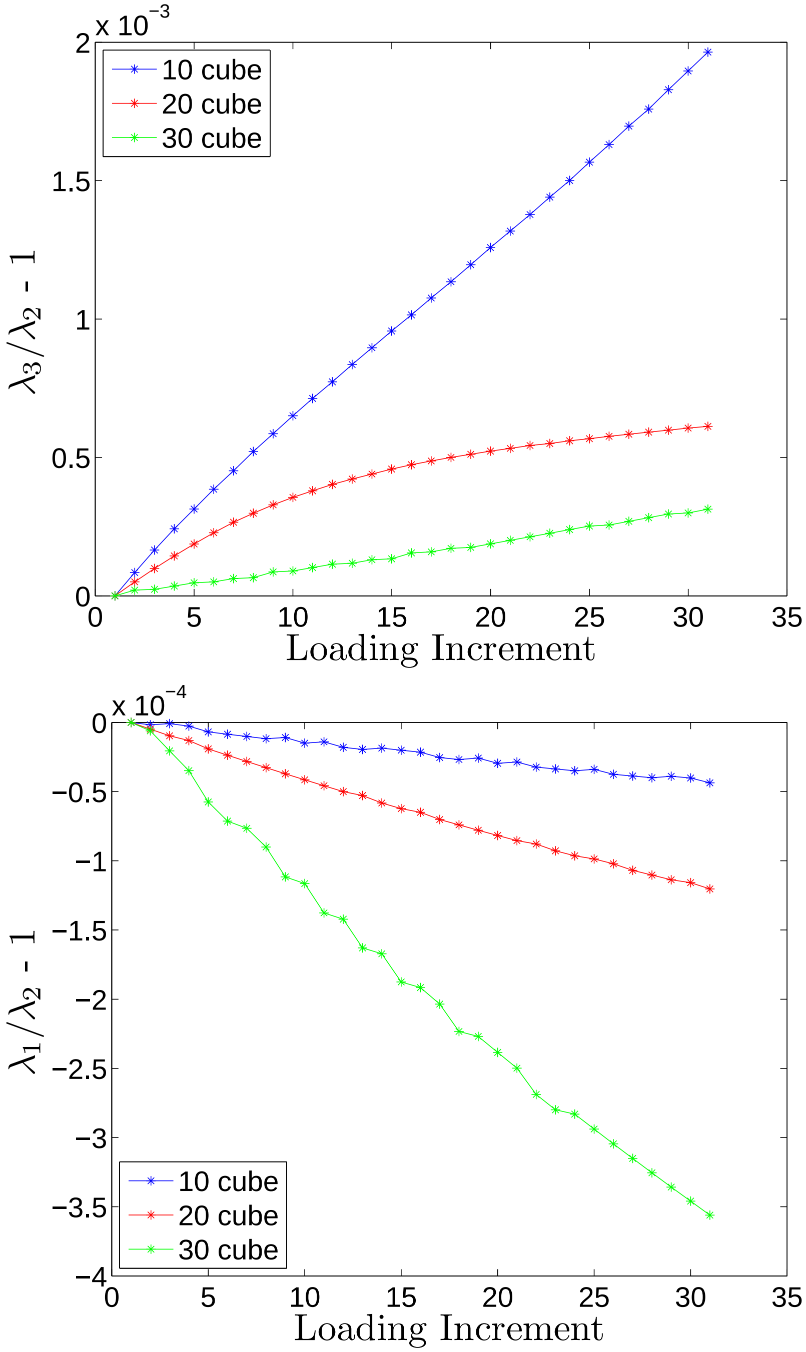


Figure 3.3: RVE's of the polymer are deformed under loading. The top curve illustrates the ratio λ_2/λ_3 for the 10, 20, and 30 cubes under uniaxial loading; if the polymer was perfectly symmetric, the ratio would be one. The bottom set of curves correspond to the cubes under biaxial loading and shows the ratio λ_1/λ_2. Again, if the RVE was perfectly symmetric, this value would be one.

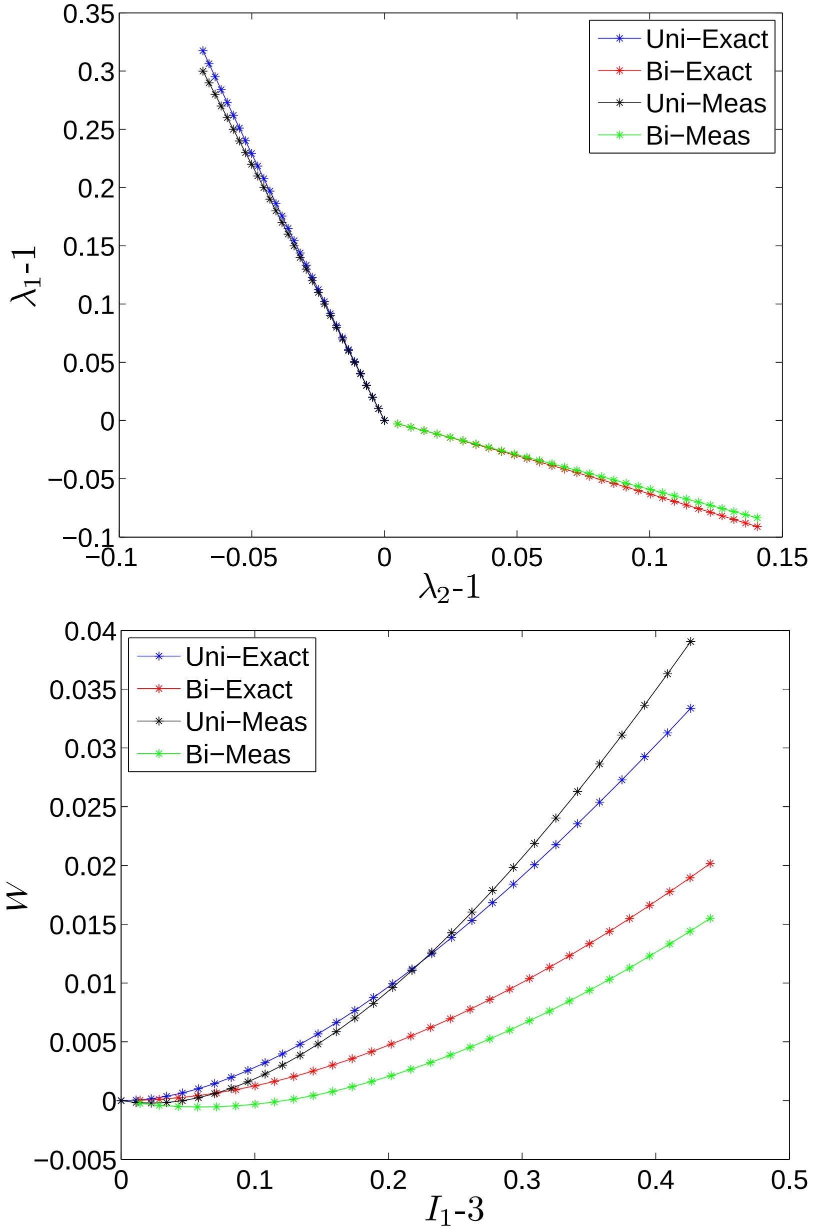


Figure 3.4: Resulting parameter fits for the Compressible Neo-Hookean material based upon measurements from a 30 × 30 × 30 polymer RVE. The top figure shows the values expected for λ_1, based on λ_2, from the exact solution to the uniaxial and biaxial extension problems compared to those measured in the numerical experiment. Similarly, the bottom figure compares the value expected from the Compressible Neo-Hookean energy density equation and the measured energy density on the RVE.

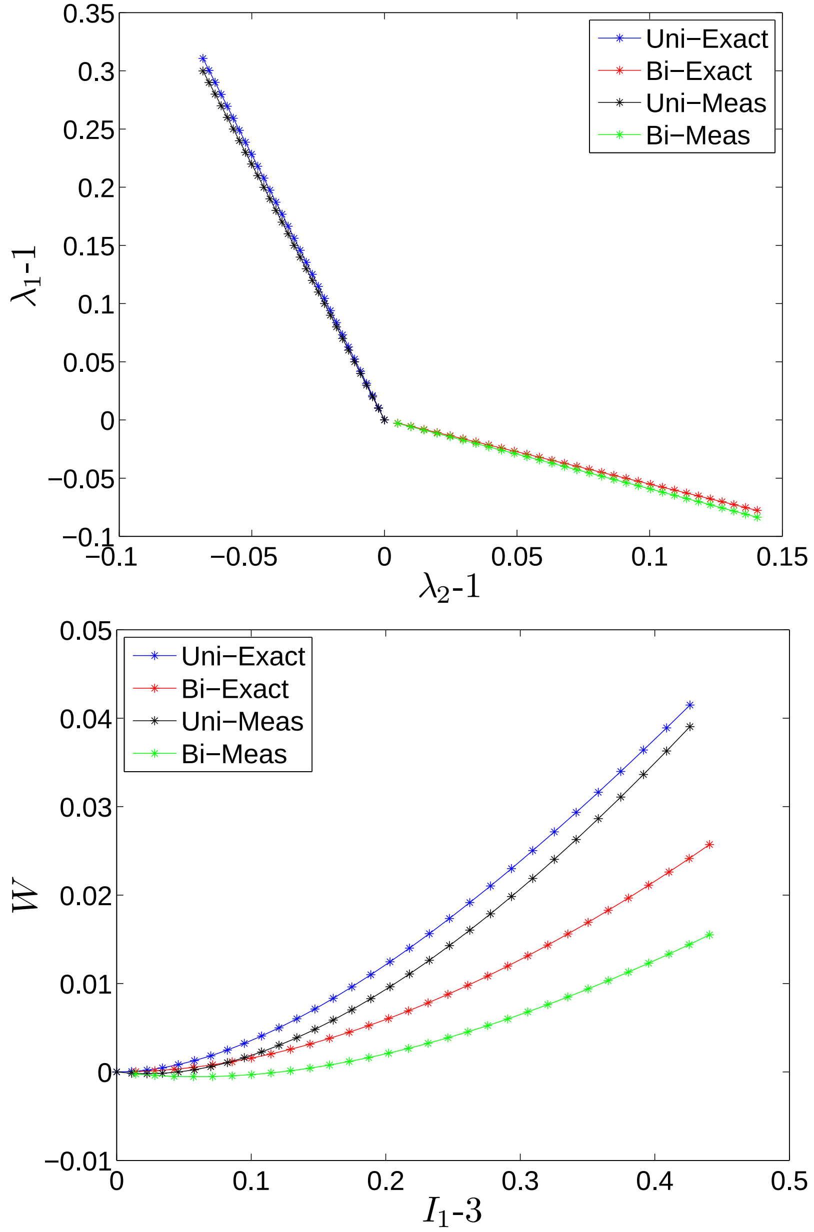


Figure 3.5: Resulting parameter fits for the Compressible Mooney-Rivlin material based upon measurements from a 30 × 30 × 30 polymer RVE. The top figure shows the values expected for λ_1, based on λ_2, from the exact solution to the uniaxial and biaxial extension problems compared to those measured in the numerical experiment. Similarly, the bottom figure compares the value expected from the Compressible Mooney-Rivlin energy density equation and the measured energy density on the RVE.

3.4 Finite Element Formulation

Let $\Omega_0 \subset \mathbb{R}^3$ be an open, connected, bounded set representing an undeformed body $\mathcal{B}$ in free space with boundary Γ. Ω_0 is referred to as the reference configuration. Let the material points of $\mathcal{B}$ be denoted by $\boldsymbol{X} \in \mathbb{R}^3$. Under the action of body forces $\boldsymbol{b}$ acting on all of Ω_0, applied tractions $\boldsymbol{t}$ acting on Γ_N, the body undergoes a motion $\boldsymbol{\varphi}(\boldsymbol{X}) : \Omega_0 \to \Omega$. Under the action of $\boldsymbol{\varphi}$, the body is mapped to the current configuration Ω.

Let $\Pi(\boldsymbol{\varphi})$ be the total energy function of body $\mathcal{B}$. Furthermore, let $W(\nabla\boldsymbol{\varphi})$ be the given strain energy density function of the body. Then,

$$\Pi(\boldsymbol{\varphi}) = \int_{\Omega_0} W(\nabla\boldsymbol{\varphi}) d\mathbf{X} - \int_{\Omega_0} \rho_0 \mathbf{b} \cdot \varphi d\mathbf{X} - \int_{\Gamma_N} \mathbf{t} \cdot \varphi d\Gamma_N \tag{3.6}$$

where $d\mathbf{X} = dX_1 dX_2 dX_3$.

The goal of the simulation is to solve the following minimization problem:

$$\text{Find } \boldsymbol{\varphi}^* \in V : \Pi(\boldsymbol{\varphi}^*) = \min_{\boldsymbol{\varphi} \in V} \Pi(\boldsymbol{\varphi}) \tag{3.7}$$

where $V = \left\{\boldsymbol{\varphi} : \overline{\Omega}_0 \to \mathbb{R}^3 : W(\nabla\boldsymbol{\varphi}) \in L^1(\Omega_0), \det \nabla\boldsymbol{\varphi} > 0 \text{ a.e. in } \Omega_0\right\}$ [24]. One of the conditions on W sufficient to allow the existence of minimizers of Π is that W be polyconvex. For polyconvexity of W, it is sufficient W be a convex function of the invariants. See [24] and the references therein. In particular, polyconvexity can be easily verified for the functions (3.5) under consideration with appropriate choices of signs of the material coefficients.

In this work, the function φ is approximated by trilinear hexahedra. The solution strategy is identical as that described in Section 2.4. Thus, the gradient and Hessian of the energy function is required and the corresponding subroutines are provided to the software packages TAO [19] and PETSc [9, 8, 10]. Throughout, repeated indices are to be summed according the the Einstein summation convention. If repeated indices are not to be summed, the indices will be underlined: $\underline{i}$. Lowercase $i, j, k, \ldots$ will denote spatial coordinate indices, while uppercase $I, J, K, \ldots$ denote material coordinate indices. The indices A, B denotes global basis function indices. Appendix E provides full details of the derivation of the gradient and Hessian and they are, respectively, given by

$$\boxed{\begin{aligned} \frac{\partial \Pi(d_{kC})}{\partial d_{iA}} &= \int_{\Omega_0} F_{\underline{i}I} S_{IJ} N^{\underline{i}}_{A,J} d\mathbf{X} - \int_{\Omega_0} \rho_0 B_{\underline{i}} N^{\underline{i}}_{A} d\mathbf{X} - \int_{\Gamma_N} T_{\underline{i}} N^{\underline{i}}_{A} d\Gamma_N \\ \frac{\partial^2 \Pi}{\partial d_{iA} \partial d_{jB}} &= \int_{\Omega_0} N^{\underline{j}}_{B,I} S_{IJ} N^{\underline{j}}_{A,J} + F_{\underline{i}I} C_{IJKL} F_{\underline{j}K} N^{\underline{i}}_{A,J} N^{\underline{j}}_{B,L} d\mathbf{X} \end{aligned}} \tag{3.8}$$

where d_{iA} are the coefficients of the shape functions approximating φ_i, S_{IJ} is the second Piola-Kirchhoff stress tensor, and C_{IJKL} is the elasticity tensor. Both $\mathbf{S}$ and $\mathbb{C}$ can be written in terms of the invariants of the deformation and are given in (E.17) and (E.18), respectively.

3.5 Formulation to Incorporate Initial Strain

This section details the augmentation of the equations of nonlinear elasticity to incorporate the inherent initial strain observed in the lattice model. To elaborate, the lattice model is composed of a network of harmonic springs. Implicit in the spring model is a reference length to which any deformation is compared. Thus, when the bonds are formed during the polymerization process and the molecules form a perfect lattice, the springs are stretched from the reference length and, therefore, the dimensions of the body shrink under the action of the stretched springs and not due to external loading. See, for example, Figure 2.9.

The difficulty here is that the equations of continuum mechanics do not inherently contain this stretch in the lattice: by default the reference configuration of the continuum body is strain free. Thus, additional information must be supplied to provide this initial strain to the continuum body. The main idea is to consider three configurations of a material body: the reference configuration Ω_0 in which the polymer assumes the prismatic form defined by the lattice boundaries after the polymerization process (generally cubic), the current configuration Ω in which boundary conditions are imposed and densification takes place (due to the initial strain), and a "strain-free" configuration $\hat{\Omega}$ in which no external forces act on the body so it is in equilibrium with zero boundary tractions. This is referred to as a *fictitious equilibrium configuration.* Let $\mathbf{X}$ denote a material point in the reference configuration and $\hat{\mathbf{X}}$ the material particle in the fictitious equilibrium configuration. These are related by a differentiable, invertible map $\hat{\varphi} : \overline{\hat{\Omega}} \to \overline{\Omega_0}$ such that $\mathbf{X} = \hat{\varphi}(\hat{\mathbf{X}})$. The spatial

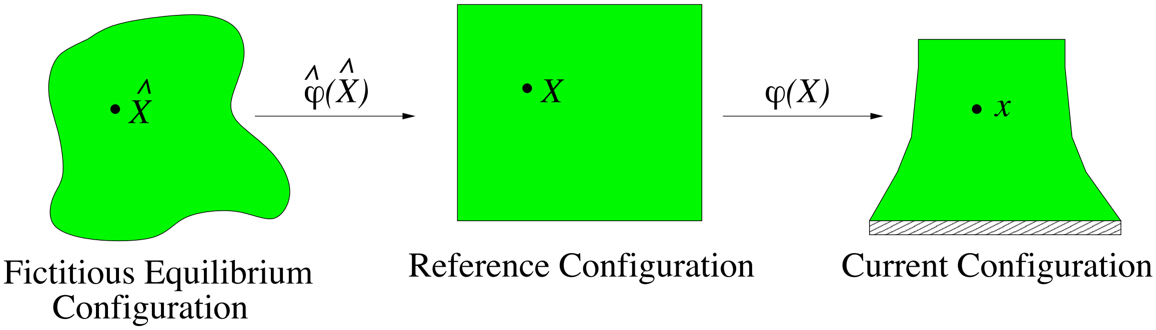


Figure 3.6: Incorporating initial strain into nonlinear elasticity.

positions of $\mathbf{x}$ in the current configuration are $\mathbf{x} = \varphi(\mathbf{X}), \varphi : \overline{\Omega_0} \rightarrow \overline{\Omega}$. Figure 3.6 represents this process schematically. Thus,

$$\mathbf{x} = \varphi \circ \hat{\varphi} = \varphi(\hat{\varphi}(\hat{\mathbf{X}})) \tag{3.9}$$

and, using the chain rule:

$$\begin{aligned} \nabla_{\hat{\mathbf{X}}}\mathbf{x} &= \nabla_{\mathbf{X}}\varphi(\mathbf{X})\nabla_{\hat{\mathbf{X}}}\hat{\varphi}(\hat{\mathbf{X}}) \\ &= \mathbf{F}\hat{\mathbf{F}} \end{aligned} \tag{3.10}$$

Therefore, the gradient $(3.8)_1$ and Hessian $(3.8)_2$ become

$$\boxed{\begin{aligned} \frac{\partial \Pi(d_{kC})}{\partial d_{iA}} &= \int_{\Omega_0} F_{\underline{i}K}\hat{F}_{KI}S_{IJ}N^{\underline{i}}_{A,M}\hat{F}_{MJ}d\mathbf{X} - \int_{\Omega_0} \rho_0 B_{\underline{i}}N^{\underline{i}}_{A}d\mathbf{X} - \int_{\Gamma_N} T_{\underline{i}}N^{\underline{i}}_{A}d\Gamma_N \\ \frac{\partial^2 \Pi}{\partial d_{iA}\partial d_{jB}} &= \int_{\Omega_0} N^{\underline{j}}_{B,K}\hat{F}_{KI}S_{IJ}N^{\underline{i}}_{A,M}\hat{F}_{MJ}d\mathbf{X} + \\ &\quad \int_{\Omega_0} F_{\underline{i}M}\hat{F}_{MI}C_{IJKL}F_{\underline{j}N}\hat{F}_{NK}N^{\underline{i}}_{A,O}\hat{F}_{OJ}N^{\underline{j}}_{B,P}\hat{F}_{PL}d\mathbf{X} \end{aligned}} \tag{3.11}$$

The task now is to supply $\hat{\mathbf{F}}$. In general, $\hat{\mathbf{F}} = \hat{\mathbf{F}}(\hat{\mathbf{X}})$, but in the present case, $\hat{\mathbf{F}}$ is the result of a homogeneous deformation of the RVE of the form:

$$\hat{\mathbf{F}} = \lambda \mathbf{I} \tag{3.12}$$

where $\mathbf{I}$ is the identity matrix and λ is a constant to be determined. The form (3.12) is justified for the case of the polymer under consideration as the bonds are random

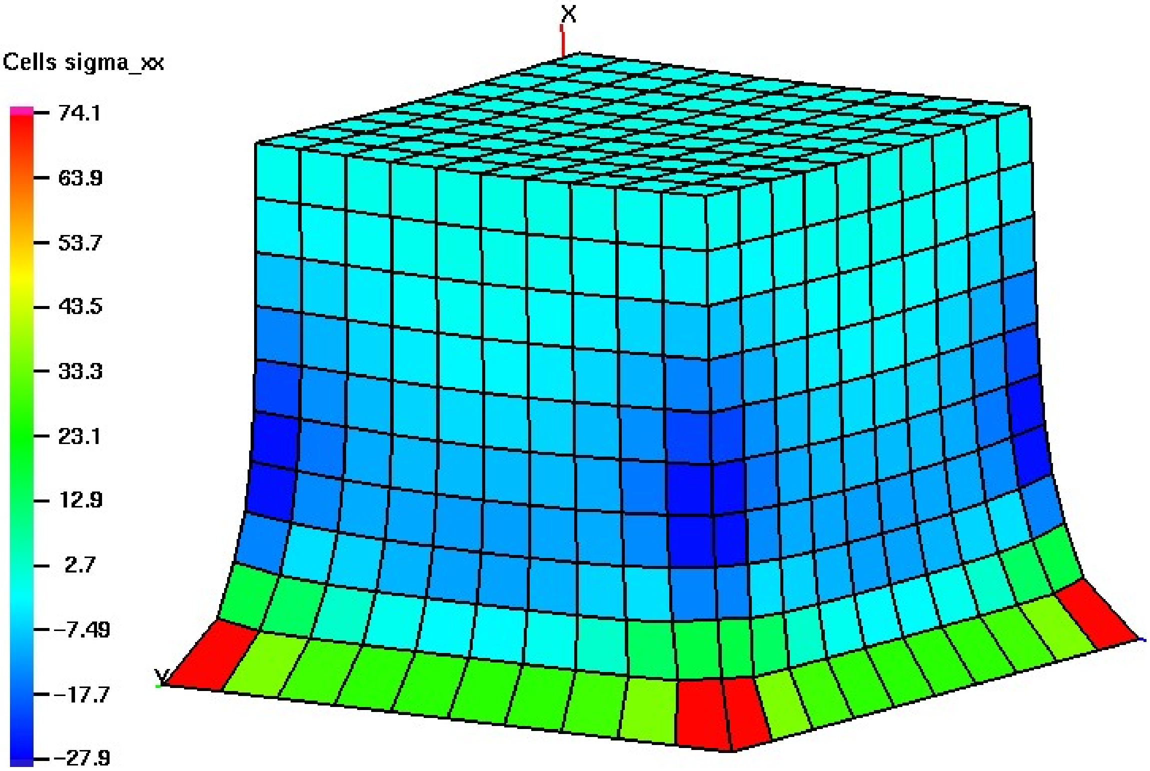


Figure 3.7: A $1,000$ element calculation of a Compressible Mooney-Rivlin material subjected to an initial strain. In this case, the parameter $\lambda = 1.25$.

in direction so that there is no preferred direction for the deformation. Therefore, numerical experiments on RVE's of the polymer material can be used to determine the constant λ. Note that, in the case of $\hat{\mathbf{F}} = \lambda \mathbf{I}$

$$\hat{\mathbf{C}} = \hat{\mathbf{F}}^T \hat{\mathbf{F}} = \lambda^2 \mathbf{I} \tag{3.13}$$

so that $\lambda^2 = \lambda_1 = \lambda_2 = \lambda_3$. Thus, the stretch of the body can be measured on an RVE of polymer when being equilibrated; that is, the stretch from the equilibrated configuration to the lattice configuration. Actually, what is measured is the inverse λ^{-1} since in the molecular RVE experiments, the lattice shrinks from the cubic reference configuration to the relaxed "fictitious equilibrium configuration". Figure 3.7 shows an example of the incorporation of initial strain into the finite element formulation.

Chapter 4

Coupling Particle and Continuum Models Using the Arlequin Method

4.1 Introduction

This chapter details the development of a general scheme to systematically couple particle and continuum models in order to produce surrogates of the molecular base model. The strategy involves the enforcing displacement and/or derivative (energy) constraints between the two models. The constraints are achieved using Lagrange multipliers on a region of overlap between the continuum and particle models. This scheme is an adaptation of the Arlequin method [18] used for coupling two continuum models with differing scales of finite element discretization (a global coarse mesh and a local fine mesh).

Section 4.2 develops a detailed mathematical analysis of a one dimensional model problem coupling a simple harmonic spring model and a linearly elastic continuum. Numerical experiments are presented illustrating the mathematical developments. This material is contained in [12]. Section 4.3 extends the formulation to the coupling of the three-dimensional, nonlinear polymer model and the nonlinear, elastic continuum model discussed previously. Some details of the solution algorithm are presented followed by several numerical experiments illustrating the behavior of the solutions.

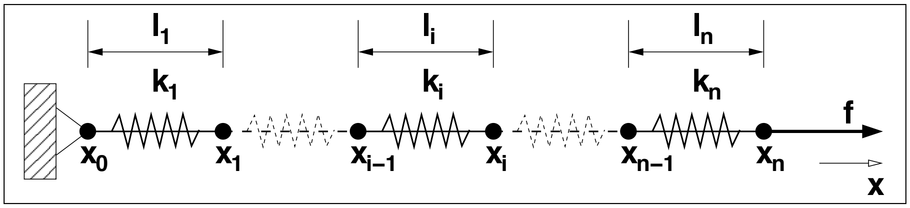


Figure 4.1: System of $n+1$ particles connected with n harmonic springs.

4.2 Mathematical Analysis of a 1-D Problem

Particle Model

The simplest one-dimensional model of a polymer chain is embodied in the system of $n+1$ particles with n harmonic springs representing covalent bonds, shown in Figure 4.1. In analogy with the characterization of a corresponding continuum model discussed in the preceding chapter, a portion of the chain is replaced by a model of an elastic rod, the properties of which are determined by homogenization methods. The goal is to develop theory and algorithms for appropriate coupling of the particle and the continuum models. To this end, let k_i denote the stiffness of the bond (the spring stiffness) between particles at position x_i and x_{i-1} and let l_i denote the equilibrium lengths of the springs $i = 1, \ldots, n$. The particle chain is fixed at x_0 and subjected to a force f applied at x_n (see Figure 4.1). The potential energy of such a system is given by

$$E_d(w) = \frac{1}{2}\sum_{i=1}^{n} k_i \left(w_i - w_{i-1}\right)^2 - f w_n \tag{4.1}$$

where w_i is the displacement of particle i along the x-direction.

The equilibrium displacement of such a system, denoted $w \in \mathbb{R}^{n+1}$, is obtained by minimizing the potential energy:

$$E_d(w) = \min_{\substack{z\in\mathbb{R}^{n+1}\\ z_0=0}} E_d(z) \tag{4.2}$$

Thus, w are stationary points of $E_d(z)$ and satisfy

$$\lim_{\theta\to 0}\frac{1}{\theta}\left(E_d(w+\theta z) - E_d(w)\right) = \langle DE_d(w), z\rangle = 0$$

In other words, the displacements w at equilibrium satisfy

$$\begin{aligned}
&\sum_{i=1}^{n} k_i \left(w_i - w_{i-1}\right)\left(z_i - z_{i-1}\right) = f z_n \quad \forall z \in \mathbb{R}^{n+1} \\
&w_0 = 0
\end{aligned} \tag{4.3}$$

Rewriting (4.3) as

$$\begin{aligned}
k_1 \left(w_1 - w_0\right) z_1 + \sum_{i=1}^{n-1} \left[k_i \left(w_i - w_{i-1}\right) - k_{i+1}\left(w_{i+1} - w_i\right)\right] z_i \\
+ k_n \left(w_n - w_{n-1}\right) z_n = f z_n \quad \forall z \in \mathbb{R}^{n+1}
\end{aligned} \tag{4.4}$$

the system of equations can be represented more compactly in matrix form as

$$z^T A w = z^T f \quad \forall z \in \mathbb{R}^{n+1} \tag{4.5}$$

where, for simplicity, $f^T = (0, \ldots, 0, f)$ and

$$A = \begin{bmatrix}
k_1 & -k_1 & 0 & 0 & 0 & \ldots & 0 \\
-k_1 & k_1 + k_2 & -k_2 & 0 & 0 & \ldots & 0 \\
0 & -k_2 & k_2 + k_3 & -k_3 & 0 & \ldots & 0 \\
 & & & \ddots & & & \\
 & & & & \ddots & & \\
0 & \ldots & \ldots & 0 & -k_{n-1} & k_{n-1} + k_n & -k_n \\
0 & \ldots & \ldots & 0 & 0 & -k_n & -k_n
\end{bmatrix} \tag{4.6}$$

It is clear that the matrix A is symmetric positive semi-definite and induces a semi-norm on $\mathbb{R}^{n+1}$:

$$|z| = \sqrt{z^T A z}$$

Indeed, $|z| = 0$ if z is a constant vector of $\mathbb{R}^{n+1}$. It then becomes a norm on the subspace $\{z \in \mathbb{R}^{n+1} : z_0 = 0\}$.

Continuum Model

One possible coarse-grain approximation of this particle model is a linear elastic rod. Then the system of springs is replaced by a continuous elastic rod in domain Ω, with

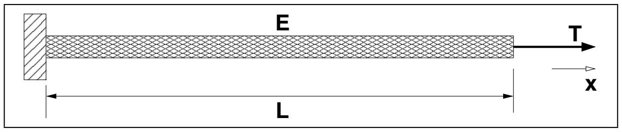


Figure 4.2: Elastic bar of length L with modulus E and loaded under traction T.

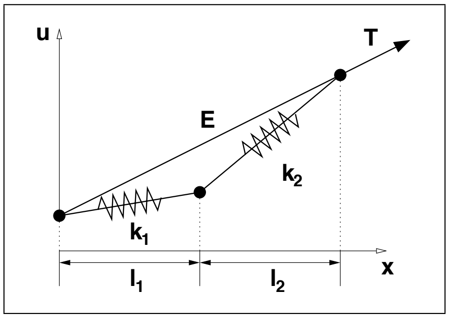


Figure 4.3: Homogenization of spring model on a representative cell.

length L, modulus E, and subjected to traction $T = f/A$, A being the cross-sectional area of the bar. The displacement in the bar is denoted u; see Figure 4.2. The material is assumed to obey Hooke's law with a stress-strain relationship $\sigma = E\varepsilon$ and $\varepsilon = u'$. The total energy of this system is thus given by

$$\begin{aligned} E_c &= \int_\Omega \frac{A}{2}\sigma(x)\varepsilon(x)\,dx - AT(L)u(L) \\ &= \int_\Omega \frac{AE}{2}\left(u'\right)^2\,dx - AT(L)u(L) \end{aligned} \tag{4.7}$$

To obtain the elastic modulus, a representative cell of springs is considered (see Figure 4.3) so that, in a system consisting of a periodic array of two springs with stiffness k_1, k_2 and equilibrium length l_1, l_2,

$$AE = \frac{k_1 k_2}{k_1 + k_2}(l_1 + l_2) \tag{4.8}$$

The modulus of elasticity E is derived here by equating the energy in the representative cell with the energy one would obtain if a linear elasticity model were used. For simplicity, it is implicitly assumed that A is unity.

As with the spring model, the equilibrium state for the continuum model is found by minimizing the energy (4.7). This minimization yields the following varia-

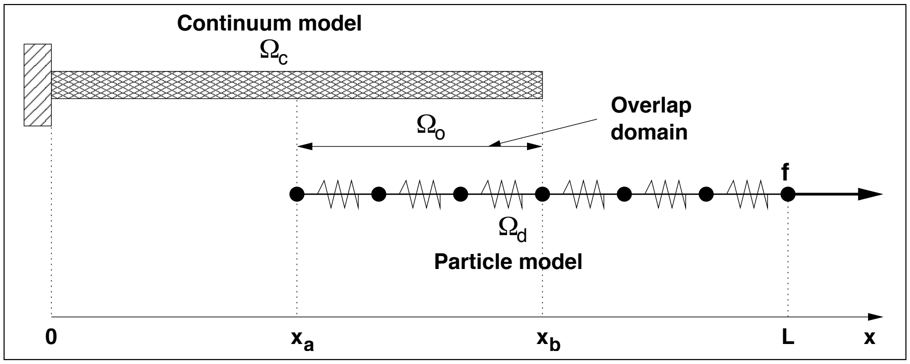


Figure 4.4: Arlequin model that replaces the particle model with a combined particle and spring model.

tional boundary value problem:

$$\boxed{\begin{array}{l}\text{Find } u \in V = \{v \in H^1(\Omega) : v(0) = 0\} \text{ such that:}\\ \displaystyle\int_\Omega E u' v' dx = T(L)v(L) \qquad \forall v \in V\end{array}} \tag{4.9}$$

Coupling Scheme

It is now assumed that the continuum model is valid in region $\Omega_c = (0, x_b)$ while the particle model prevails in domain $\Omega_d = (x_a, L)$ such that $\Omega = \Omega_c \bigcup \Omega_d$ and $\Omega_o = \Omega_c \bigcap \Omega_d = (x_a, x_b)$, $|\Omega_o| \neq 0$; Ω_o is referred to as the overlap region. Let $|\Omega_c|$, $|\Omega_d|$, and $|\Omega_o|$ be the length of domains Ω_c, Ω_d, and Ω_o, respectively. The particle model has thus been reduced from $n+1$ to $m+1$ particles that are connected by m harmonic springs , supposedly with $m \ll n$ in practice. See Figure 4.4.

The main idea of the Arlequin method is to modify the energies as follows:

$$\begin{aligned}\hat{E}_c &= \int_{\Omega_c} \alpha_c(x) \frac{E}{2} \left(u'\right)^2 \, dx \\ \hat{E}_d &= \frac{1}{2} \sum_{i=1}^{m} \alpha_i k_i \left(w_i - w_{i-1}\right)^2 - f w_m\end{aligned} \tag{4.10}$$

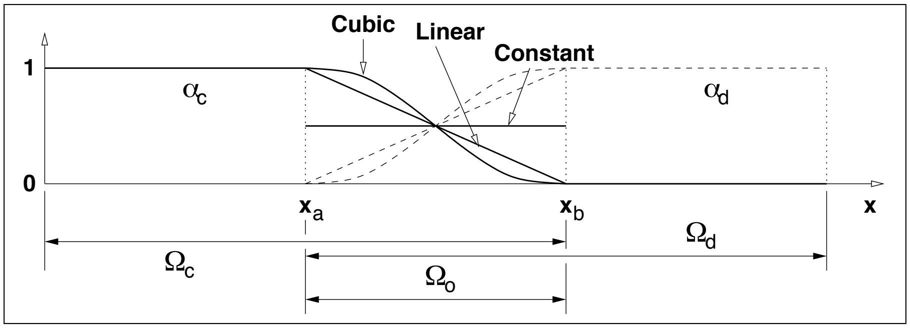


Figure 4.5: Plot of different functions used for α_c and α_d.

where weighting coefficients α_i and α_c are introduced such that

$$\begin{aligned} &\alpha_c(x) + \alpha_d(x) = 1 \quad \forall x \in \Omega \\ &\alpha_c(x) = \begin{cases} 1 & \forall x \in \Omega_c \setminus \Omega_o \\ 0 & \forall x \in \Omega_d \setminus \Omega_o \end{cases} \\ &\alpha_i = \alpha_d \left(\frac{1}{2} (x_i + x_{i-1}) \right), \quad i = 1, \ldots, m \end{aligned} \tag{4.11}$$

In the overlap region Ω_o, the coefficient α_c (and thus α_d) can be chosen in different ways. Some intuitive and apparently attractive candidates are (see Figure 4.5):

$$\begin{aligned} \alpha_c(x) &= \frac{1}{2} \quad \forall x \in \Omega_o \\ \alpha_c(x) &= 1 - \frac{(x - x_a)}{x_b - x_a} \quad \forall x \in \Omega_o \\ \alpha_c(x) &= \frac{-(x - x_b)^2 (2x - 3x_a + x_b)}{(x_a - x_b)^3} \quad \forall x \in \Omega_o \end{aligned} \tag{4.12}$$

where x_a and x_b denote the left and right end point of Ω_o.

In the overlap region, the main idea is to constrain the displacements u and w to be equal in some appropriate measure. The first step is to convert the discrete displacements w into a displacement field Πw that can be compared to u on Ω_o. The natural way to do this is to take Π as the linear interpolation operator. Other interpolation schemes are possible, but only the linear interpolant is considered in the present work.

Thus, the "energy" generated by the mismatch of u and Πw on Ω_o is

$$\|u - \Pi w\|^2 = \int_{\Omega_o} \beta_1 \left(u - \Pi w\right)^2 + \beta_2 \left(u - \Pi w\right)'^2 \; dx \tag{4.13}$$

where (β_1, β_2) are non-negative weight parameters. These can also be chosen so as to scale the two terms in the integral. For example, $(\beta_1, \beta_2) = (1, 0)$ characterizes the square of the L^2 norm of the difference between u and the interpolant, $(\beta_1, \beta_2) = (0, 1)$ the square of the H^1 seminorm, and $(\beta_1, \beta_2) = (1, 1)$ the square of the H^1 norm on Ω_o.

The coupled problem consists of finding u and w, in appropriate spaces V_c and V_d, respectively (defined below), that minimize the total energy and satisfy the constraint $\|u - \Pi w\| = 0$, i.e.

$$\hat{E}(u, w) = \hat{E}_d(w) + \hat{E}_c(u) = \min_{\substack{v \in V_c, z \in V_d \\ \|v - \Pi z\| = 0}} \left(\hat{E}_d(z) + \hat{E}_c(v)\right) \tag{4.14}$$

Introducing the coupling term,

$$b\left(\lambda, (u, w)\right) = \int_{\Omega_o} \beta_1 \lambda \left(u - \Pi w\right) + \beta_2 \lambda' \left(u - \Pi w\right)' \; dx \tag{4.15}$$

the minimization problem (4.14) can be recast into the following saddle point problem:

$$\min_{v \in V_c, z \in V_d} \max_{\mu \in M} \left(\hat{E}_d(z) + \hat{E}_c(v) + b\left(\mu, (v, z)\right)\right) \tag{4.16}$$

where M is an appropriate space for the Lagrange multipliers.

4.2.1 Mathematical Analysis of the Coupled Formulation

Let $V_c = \{v \in H^1(\Omega_c) : v(0) = 0\}$ and $V_d = \{z \in \mathbb{R}^{m+1}\}$ be the vector spaces of test functions for the continuum and discrete models, respectively, and let Π be the linear interpolant $\Pi : V_d \to H^1(\Omega_o)$. In what follows, a function $v \in V_c$ will not be distinguished from its restriction to the space $H^1(\Omega_o)$. The vector space for the

Lagrange multipliers is defined as

$$M = \begin{cases} H^1(\Omega_o)/\mathbb{R}, & \text{if } \beta_1 = 0 \\ H^1(\Omega_o), & \text{otherwise} \end{cases} \tag{4.17}$$

with associated norm:

$$\|\mu\|_M = \sqrt{\int_{\Omega_o} \beta_1 \mu^2 + \beta_2 \mu'^2 dx}$$

Let the average of z on Ω_o be denoted:

$$\bar{z} = \sum_{i=1}^{n_o} \frac{l_i}{|\Omega_o|} \frac{z_i + z_{i-1}}{2}$$

where n_o is the number of springs on Ω_o. *The restrictive assumption made here is that the overlap region exactly coincides with a given set of complete springs. In other words, the domain Ω_o is not allowed to cover only part of a spring.* The seminorm $|\cdot|_{V_d}$ on V_d is defined as

$$|z|_{V_d} = \sqrt{\sum_{i=1}^{m} k_i (z_i - z_{i-1})^2}$$

The norms on V_c and V_d are then chosen as:

$$\begin{aligned} \|v\|_{V_c} &= \sqrt{\int_{\Omega_c} E|v'|^2 dx} \\ \|z\|_{V_d} &= \sqrt{|z|_{V_d}^2 + \delta \bar{z}^2} \end{aligned} \tag{4.18}$$

where δ is a dimensionally consistent weighting constant defined below.

The product space $X = V_c \times V_d$ is defined with pairs of X denoted, for example, as $U = (u, w)$, $V = (v, z)$, and with norm:

$$\|V\|_X = \sqrt{\|v\|_{V_c}^2 + \|z\|_{V_d}^2} \tag{4.19}$$

and the kernel space of $b(\cdot, \cdot)$ is defined as the subspace of X such that:

$$X_0 = \{V \in X : b(\mu, V) = 0 \quad \forall \mu \in M\} \tag{4.20}$$

Then the problem to be solved is the following saddle point problem:

$$\begin{aligned}&\text{Find } U \in X,\ \lambda \in M \text{ such that:}\\ &L(U,\lambda) = \inf_{V\in X}\sup_{\mu\in M} L\left(V,\mu\right)\end{aligned} \tag{4.21}$$

where the Lagrangian reads:

$$\begin{aligned}
L\left(V,\mu\right) &= \frac{1}{2}a\left(V,V\right) + b\left(\mu,V\right) - l(V)\\
a\left(U,V\right) &= \int_{\Omega_c} \alpha_c E u' v' dx + \sum_{i=1}^{m} \alpha_i k_i \left(w_i - w_{i-1}\right)\left(z_i - z_{i-1}\right)\\
b\left(\mu,V\right) &= \int_{\Omega_o} \beta_1 \mu(v - \Pi z) + \beta_2 \mu'(v - \Pi z)' dx\\
l(V) &= f z_m
\end{aligned} \tag{4.22}$$

The saddle point problem (4.21) can thus be recast as:

$$\boxed{\begin{array}{l}\text{Find } U \in X,\ \lambda \in M \text{ such that:}\\ \begin{array}{rcll} a(U,V) + b(\lambda,V) &=& l(V) & \forall V \in X\\ b(\mu,U) &=& 0 & \forall \mu \in M\end{array}\end{array}} \tag{4.23}$$

Problem (4.23) is well posed for $\beta_1 \geq 0$ and $\beta_2 > 0$. This result immediately follows from results in Ben Dhia and Rateau [16]. Nevertheless, a detailed proof is given here with the main objective of explicitly deriving the constants associated with the problem in order to study the influence the geometrical and material parameters, the coupling parameters β_1 and β_2, and the length of the overlap domain on the coupled solutions. Proofs of continuity of the forms $a(\cdot,\cdot)$, $b(\cdot,\cdot)$, and $l(\cdot)$ are relatively straightforward and provided for completeness in Appendix F.2. It is shown below that $a(\cdot,\cdot)$ is coercive and that the coupling term $b(\cdot,\cdot)$ satisfies the Babuška-Brezzi condition [5, 20]. Technical lemmas are presented in Appendix F.1.

Lemma 4.1. ***(Coercivity of*** a***)*** *Let* α_c *and* α_d *be constant or linear functions defined by* $(4.12)_1$ *and* $(4.12)_2$*. Then, with above notation and definitions, there exists a* *constant* $\gamma_a > 0$ *such that:*

$$\inf_{U \in X_0} \sup_{V \in X_0} \frac{|a\,(U,V))\,|}{\|U\|_X \|V\|_X} > \gamma_a$$
$$\sup_{U \in X_0} |a\,(U,V)\,| > 0 \quad \forall V \in X_0, V \neq 0$$

with

$$\gamma_a = \begin{cases} \gamma_1 & \text{if } \beta_1 = 0 \\ \gamma_1 \min\left(\dfrac{1}{2}, \dfrac{E|\Omega_o|}{\delta|\Omega_c|^2}\right) & \text{if } \beta_1 > 0 \end{cases}$$

and

$$\gamma_1 = \begin{cases} \dfrac{1}{2} & \alpha_c = \alpha_d = \dfrac{1}{2} \\ \dfrac{1}{2} \min\limits_i \left(\dfrac{E}{k_i l_i}, \dfrac{k_i l_i}{E}\right) & \alpha_c, \alpha_d \text{ linear} \end{cases}$$

Proof. It suffices to show that $a(\cdot,\cdot)$ is coercive on X_0. Let $V = (v,z) \in X_0$. It is shown that

$$a(V,V) \geq \gamma_1 \left(\|v\|_{V_c}^2 + |z|_{V_d}^2\right)$$

where γ_1 is a constant that depends on α_c and α_d being constant or linear.

By definition of the bilinear form, and the fact that $\alpha_c = 1$ on $\Omega_c \setminus \Omega_o$ and $\alpha_d = 1$ on $\Omega_d \setminus \Omega_o$:

$$\begin{aligned} a(V,V) &= \int_{\Omega_c} \alpha_c E(v')^2 \, dx + \sum_{i=1}^{m} \alpha_i k_i (z_i - z_{i-1})^2 \\ &= \int_{\Omega_c \setminus \Omega_o} E(v')^2 \, dx + \sum_{i=n_o+1}^{m} k_i (z_i - z_{i-1})^2 + \int_{\Omega_o} \alpha_c E(v')^2 \, dx + \sum_{i=1}^{n_o} \alpha_i k_i (z_i - z_{i-1})^2 \end{aligned}$$

First, the overlap terms are divided in half:

$$\begin{aligned} \int_{\Omega_o} \alpha_c E(v')^2 \, dx + \sum_{i=1}^{n_o} \alpha_i k_i (z_i - z_{i-1})^2 =& \frac{1}{2}\left(\int_{\Omega_o} \alpha_c E(v')^2 \, dx + \sum_{i=1}^{n_o} \alpha_i k_i (z_i - z_{i-1})^2\right) \\ &+ \frac{1}{2}\left(\int_{\Omega_o} \alpha_c E(v')^2 \, dx + \sum_{i=1}^{n_o} \alpha_i k_i (z_i - z_{i-1})^2\right) \end{aligned}$$

Next, the continuum term and the discrete term are examined and it is evident that they should be recombined. The fact that X_0 consists of functions v and vectors z

such that $v = \Pi z$ on Ω_o (and therefore $v' = (\Pi z)'$) is used, and

$$(\Pi z)' = \sum_{i=1}^{n_o} \frac{z_i - z_{i-1}}{l_i}$$

Thus,

$$\begin{aligned}
\frac{1}{2}\int_{\Omega_o} \alpha_c E(v')^2 \, dx &= \frac{1}{2}\sum_{i=1}^{n_o} E \int_{x_{i-1}}^{x_i} \alpha_c \left(\frac{z_i - z_{i-1}}{l_i}\right)^2 \, dx \\
&= \frac{1}{2}\sum_{i=1}^{n_o} E \left(\frac{z_i - z_{i-1}}{l_i}\right)^2 \int_{x_{i-1}}^{x_i} \alpha_c \, dx \\
&\geq \frac{1}{2}\min_i \left(\frac{E}{k_i l_i}\right) \sum_{i=1}^{n_o} (1-\alpha_i)\, k_i (z_i - z_{i-1})^2
\end{aligned}$$

Then, repeating the same procedure in opposite order on the discrete term gives:

$$\frac{1}{2}\sum_{i=1}^{n_o} \alpha_i k_i (z_i - z_{i-1})^2 \geq \frac{1}{2}\min_i \left(\frac{k_i l_i}{E}\right) \int_{\Omega_o} \alpha_d E(v')^2 \, dx$$

Substituting the previous two expressions into the original expression and using the fact that $\alpha_c + \alpha_d = 1$ gives

$$\begin{aligned}
a(V,V) \geq & \int_{\Omega_c \setminus \Omega_o} E(v')^2 \, dx + \sum_{i=n_o+1}^{m} k_i (z_i - z_{i-1})^2 \\
& + \frac{1}{2}\min\left(1, \min_i \left(\frac{E}{k_i l_i}\right)\right) \sum_{i=1}^{n_o} k_i (z_i - z_{i-1})^2 \\
& + \frac{1}{2}\min\left(1, \min_i \left(\frac{k_i l_i}{E}\right)\right) \int_{\Omega_o} E(v')^2 \, dx \\
\geq & \ \gamma_1 \left(\|v\|_{V_c}^2 + |z|_{V_d}^2\right)
\end{aligned}$$

and

$$\gamma_1 = \frac{1}{2}\min_i \left(\frac{E}{k_i l_i}, \frac{k_i l_i}{E}\right)$$

If $\beta_1 = 0$, the result is immediate with $\gamma_a = \gamma_1$. If β_1 is nonzero, it is observed that the term $|z|_{V_d}$ vanishes for all constant vectors z in V_d. Applying Poincaré's inequality (cf. Lemma F.1) yields

$$\begin{aligned} a(V,V) &\geq \gamma_1 \left(\frac{1}{2} \|v\|_{V_c}^2 + \frac{1}{2} \|v\|_{V_c}^2 + |z|_{V_d}^2 \right) \\ &\geq \gamma_1 \left(\frac{1}{2} \|v\|_{V_c}^2 + \frac{E}{|\Omega_c|^2} \|v\|_{L^2(\Omega_c)}^2 + |z|_{V_d}^2 \right) \end{aligned}$$

Using Lemma F.2, the fact that X_0 consists of those functions v and vectors z such that $v = \Pi z$, which implies $\overline{v} = \overline{\Pi z}$, then, observing that

$$\|v\|_{L^2(\Omega_c)}^2 \geq \|v\|_{L^2(\Omega_o)}^2 \geq \bar{v}^2 |\Omega_o| = \overline{z}^2 |\Omega_o|$$

it follows that

$$\begin{aligned} a(V,V) &\geq \gamma_1 \left(\frac{1}{2} \|v\|_{V_c}^2 + |z|_{V_d}^2 + \frac{E|\Omega_o|}{\delta |\Omega_c|^2} \delta \overline{z}^2 \right) \\ &\geq \gamma_a \|V\|_X^2 \end{aligned}$$

where

$$\gamma_a = \gamma_1 \min \left(\frac{1}{2}, \frac{E|\Omega_o|}{\delta |\Omega_c|^2} \right)$$

The proof for the case $\alpha_c = \alpha_d = 1/2$ is the same as above and the constant $\gamma_a = \gamma_1 = 1/2$. This completes the proof. □

Remark 4.1. Although the strong condition $v = \Pi z$ is used in the second part of the proof, the weaker condition $\overline{v} = \overline{\Pi z}$ could have been used. This becomes important in the proof of discrete coercivity that is addressed later.

Remark 4.2. The case in which α_c, α_d are cubic functions $(4.12)_3$ has not been investigated. It is believed that this case yields coercivity and could be proven with more sophisticated techniques.

Lemma 4.2. ***(Inf-sup condition for** b**)*** *Let* $\beta_2 > 0$. *Then, with the above notation and definitions in place, there exists a constant* $\gamma_b > 0$ *such that:*

$$\inf_{\mu \in M} \sup_{V \in X} \frac{|b(\mu, V)|}{\|\mu\|_{M_1} \|V\|_X} > \gamma_b \tag{4.24}$$

with

$$\gamma_b = \begin{cases} \sqrt{\dfrac{\beta_2}{E}} & \beta_1 = 0 \\ \sqrt{\dfrac{\beta_2}{E}} \min\left(\sqrt{\dfrac{\beta_1|\Omega_o|E}{2\delta\beta_2}}, \sqrt{\dfrac{E}{E+\delta|\Omega_o|}}\right) & \beta_1 > 0 \end{cases}$$

Proof. This proof is similar to that given in [16] for coupling two continuum models, but this proof relies upon the fact that a particle and continuum model are being coupled and, as a result, the constant γ_b is directly computed. It is sufficient to show that

$$\sup_{V\in X} \frac{|b\left(\mu, V\right)|}{\|V\|_X} > \gamma_b \|\mu\|_M \quad \forall \mu \in M$$

Since $\mu \in M$, $\mu(x_a)$ is well defined and denoted by μ_a. Let $\hat{\mu} = \mu - \mu_a$. Introduce the extension operator $S(\mu) : \mu \in M \to \hat{v} \in V_c$ such that $\hat{v} = \hat{\mu}$ on Ω_o, and $\hat{v} = 0$ on $\Omega_c \setminus \Omega_o$. Furthermore, let $\hat{z}$ be the constant vector $\hat{z} = \mu_a$. Thus, taking $\hat{V} = (\hat{v}, \hat{z})$, it is observed that

$$\sup_{V\in X} \frac{|b\left(\mu, V\right)|}{\|V\|_X} \geq \frac{\left|b\left(\mu, \hat{V}\right)\right|}{\|\hat{V}\|_X} = \frac{\|\mu\|_M^2}{\|\hat{V}\|_X}$$

It suffices to show that $\|\mu\|_M/\|\hat{V}\|_X$ is greater than a positive constant independent of μ.

Using the definition of $\|\cdot\|_X$,

$$\begin{aligned} \|\hat{V}\|_X^2 &= \int_{\Omega_o} E(\mu')^2\, dx + \delta\mu_a^2 \\ &= E|\mu|_{H^1(\Omega_o)}^2 + \delta\mu_a^2 \end{aligned}$$

Thus, if $\beta_1 = 0$, one can fix $\mu_a = 0$, and

$$\|\hat{V}\|_X^2 = E\int_{\Omega_o} \mu'^2 dx = \frac{E}{\beta_2}\|\mu\|_M^2$$

The inf-sup constant $\gamma_b = \sqrt{\beta_2/E}$.

If $\beta_1 > 0$, μ_a can be bounded in terms of $\|\mu\|_{L^2(\Omega_o)}$ and $|\mu|_{H^1(\Omega_o)}^2$. Using the

Poincaré inequality (since $\hat{\mu} = 0$ at $x = x_a$),

$$\begin{aligned}\int_{\Omega_o} \mu_a^2 dx = \int_{\Omega_o} (\mu - \hat{\mu})^2 dx &\leq 2 \int_{\Omega_o} \mu^2 + \hat{\mu}^2 dx \\ &\leq 2\|\mu\|^2_{L^2(\Omega_0)} + |\Omega_o|^2 |\hat{\mu}|^2_{H^1(\Omega_o)} \\ &\leq 2\|\mu\|^2_{L^2(\Omega_0)} + |\Omega_o|^2 |\mu|^2_{H^1(\Omega_o)}\end{aligned}$$

Since

$$\int_{\Omega_o} \mu_a^2 dx = |\Omega_o| \mu_a^2$$

the following inequality holds:

$$\mu_a^2 \leq \frac{2}{|\Omega_o|} \|\mu\|^2_{L^2(\Omega_0)} + |\Omega_o| |\mu|^2_{H^1(\Omega_o)}$$

Thus, substituting the bound for μ_a, it is concluded that

$$\begin{aligned}\|\hat{V}\|^2_X &\leq \frac{2\delta}{|\Omega_o|} \|\mu\|^2_{L^2(\Omega_o)} + (E + \delta|\Omega_o|) \, |\mu|^2_{H^1(\Omega_o)} \\ &\leq \max\left(\frac{2\delta}{\beta_1 |\Omega_o|}, \frac{E + \delta|\Omega_o|}{\beta_2} \right) \|\mu\|^2_M\end{aligned}$$

and, therefore,

$$\gamma_b = \begin{cases} \sqrt{\dfrac{\beta_2}{E}} & \beta_1 = 0 \\ \sqrt{\dfrac{\beta_2}{E}} \min\left(\sqrt{\dfrac{\beta_1 |\Omega_o| E}{2\delta\beta_2}}, \sqrt{\dfrac{E}{E + \delta|\Omega_o|}} \right) & \beta_1 > 0 \end{cases}$$

and the proof is complete. □

Remark 4.3. The result does not hold for the case in which $\beta_2 = 0$. Indeed, M would be simply the space $L^2(\Omega_0)$ and the extension operator $S(\lambda)$ is not defined in this case. This stems from the fact that the space $L^2(\Omega_o)$ is not contained in $H^1(\Omega)$.

From the continuity and coercivity of $a(\cdot, \cdot)$, from the continuity of $l(\cdot)$, and from the continuity and inf-sup condition of $b(\cdot, \cdot)$ (see Lemmas F.1, F.2, F.3, 4.1, and 4.2), we have the following theorem.

Theorem 4.1. *Let $\beta_1 \geq 0$ and $\beta_2 > 0$. Then, problem (4.23) is well-posed, in the sense that it admits a unique solution and that the solution depends continuously on*

Table 4.1: Constants from continuity conditions.

M_a	1
M_b	$\sqrt{2}\max\left(\sqrt{\frac{\beta_1\|\Omega_c\|^2+2\beta_2}{2E}}, \sqrt{\frac{\beta_1\|\Omega_o\|}{\delta}}, \sqrt{\frac{\beta_1\|\Omega_o\|^2+2\beta_2}{2\min_i k_i l_i}}\right)$
M_l	$2\|f\|\max\left(\frac{1}{\sqrt{\delta}}, \frac{1}{\sqrt{\min_i k_i}}\right)$

Table 4.2: Constants from coercivity and B-B conditions for the case $\beta_1 = 0$.

γ_a	$\frac{1}{2}\min_i\left(\frac{E}{k_i l_i}, \frac{k_i l_i}{E}\right)$
γ_b	$\sqrt{\frac{\beta_2}{E}}$

the data.

Finally, the constants obtained from continuity, coercivity, and the Babuška-Brezzi condition are summarized in Tables 4.1 – 4.3. To obtain optimality with respect to the constants, specific values are chosen for β_1, β_2, and δ. In particular, it is necessary that β_1, β_2, and δ be dimensionally consistent while also optimizing the continuity constants (i.e. not depending on the size of the domains). Table 4.4 summarizes the choice for the parameters β_1, β_2, and δ and Table 4.5 shows the resulting "optimized" constants.

Remark 4.4. Note that the constants M_b and γ_d in Table 4.5 are bounded above and below, respectively, by observing that $|\Omega_o| \leq |\Omega_c|$. Then:

$$\begin{aligned} M_b &\leq \sqrt{2}\max\left(1, \sqrt{\frac{E}{\min_i k_i l_i}}\right) \\ \gamma_b &\geq \frac{1}{2} \end{aligned} \tag{4.25}$$

However, the constant M_l increases as $|\Omega_o|$ decreases.

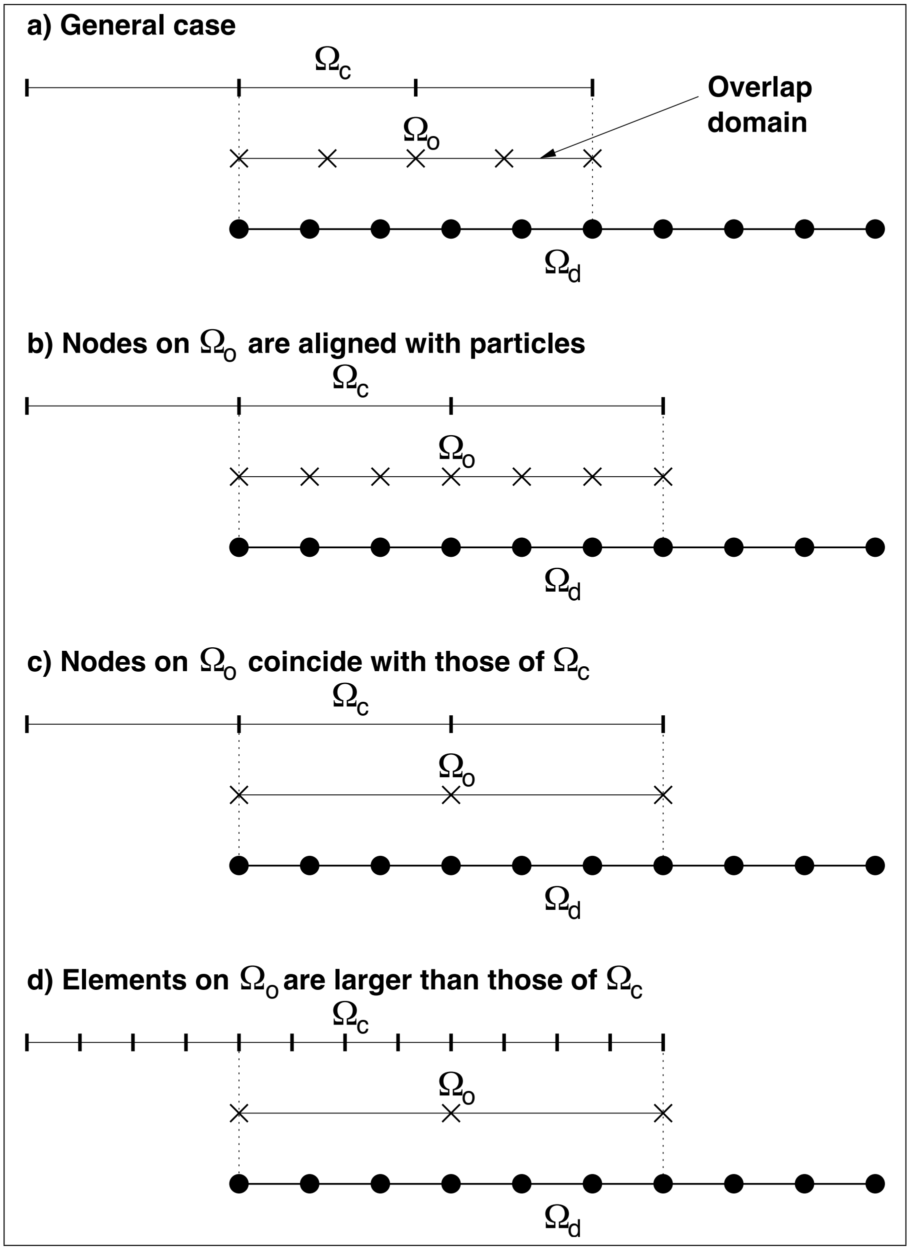


Figure 4.6: Finite element discretization of Ω_c and Ω_0 (▌ = nodes on Ω_c, ✕ = nodes on Ω_d, ● = particles on Ω_d).

Table 4.3: Constants from coercivity and B-B conditions for the case $\beta_1 > 0$.

γ_a	$\frac{1}{2}\min_i\left(\frac{E}{k_i l_i}, \frac{k_i l_i}{E}\right)\min\left(\frac{1}{2}, \frac{E\|\Omega_o\|}{\delta\|\Omega_c\|^2}\right)$
γ_b	$\sqrt{\frac{\beta_2}{E}}\min\left(\sqrt{\frac{\beta_1\|\Omega_o\|E}{2\delta\beta_2}}, \sqrt{\frac{E}{E+\delta\|\Omega_o\|}}\right)$

Table 4.4: Choice for the parameters β_1, β_2, and δ

β_1	$\frac{2E}{\|\Omega_c\|^2}$
β_2	E
δ	$\frac{E\|\Omega_o\|}{\|\Omega_c\|^2}$

4.2.2 Discrete Formulation of the Coupled Model

Let V_c^h and M^h be finite element subspaces of the vector spaces V_c and M, respectively, and let X^h be the product space $X^h = V_c^h \times V_d$. More precisely, the subspace V_c^h consists of piecewise linear continuous functions defined by the set of nodes $x_i = ih$, $i = 0, \ldots, N^e$, where N^e denotes the number of elements in the mesh. For the subspace M^h, there are several choices since the elements associated with V_c^h and M^h do not have to match (case (a) in Figure 4.6). However, for the sake of simplicity here, only three special cases are considered for M^h (see Figure 4.6, cases (b), (c), and (d)):

1. **Particle coupling:** Each node of the mesh associated with M^h coincides with the position of one particle on Ω_o and vice-versa (case (b) in Figure 4.6).

2. **Continuum coupling:** The elements of the mesh associated with M^h are exactly identical with those of V_c^h on Ω_o (case (c) in Figure 4.6).

3. **RVE coupling:** The element size h for the continuum solution is chosen arbitrarily from the equilibrium length l of the particles, but the elements for M^h are equal to the size, denoted ε, of the representative volume element (RVE) (case (d) in Figure 4.6). The continuum coupling can then be viewed as a subcase of this case.

Table 4.5: Rescaled constants for continuity, coercivity, and B-B stability for the case $\beta_1 > 0$.

M_a	1
M_b	$\sqrt{2}\max\left(1, \sqrt{\frac{E}{2\min_i k_i l_i}\left(1+\frac{\lvert\Omega_o\rvert^2}{\lvert\Omega_c\rvert^2}\right)}\right)$
M_l	$2\lvert f\rvert\max\left(\frac{\lvert\Omega_c\rvert}{\sqrt{E\lvert\Omega_o\rvert}}, \frac{1}{\sqrt{\min_i k_i}}\right)$
γ_a	$\frac{1}{4}\min_i\left(\frac{E}{k_i l_i}, \frac{k_i l_i}{E}\right)$
γ_b	$\sqrt{\frac{\lvert\Omega_c\rvert^2}{\lvert\Omega_c\rvert^2+\lvert\Omega_o\rvert^2}}$

Finally, let $U_h = (u_h, w_h)$ and $V_h = (v_h, z)$ and introduce the space X_0^h:

$$X_0^h = \left\{V_h \in X^h : b\left(\mu_h, V_h\right) = 0 \quad \forall \mu_h \in M^h\right\} \tag{4.26}$$

Then, problem (4.23) is approximated as follows:

$$\boxed{\begin{array}{l} \text{Find } U_h \in X^h,\ \lambda_h \in M^h \text{ such that:} \\ \begin{array}{rcll} a\left(U_h, V_h\right) + b\left(\lambda_h, V_h\right) & = & l(V_h) & \forall V_h \in X^h \\ b\left(\mu_h, U_h\right) & = & 0 & \forall \mu_h \in M^h \end{array} \end{array}} \tag{4.27}$$

Remark 4.5. Although V_d is a finite-dimensional space and, consequently does not need to be discretized using finite elements, the notation w_h is used to denote the solution of the particle model in (4.27) to emphasize that w_h indirectly depends on the choice of V_c^h and M^h.

Existence and Uniqueness of Solutions

In this section, it is proven that the discretized Problem (4.27) is well-posed. The lemmas of the previous section are reviewed in order to highlight the differences between the "continuous" and "discrete" problems. Consideration of continuity of $a(\cdot,\cdot)$, $b(\cdot,\cdot)$, and $l(\cdot)$ is omitted as they follow trivially (since $X^h \subset X$ and $M^h \subset M$).

One difficulty in analyzing the discretized saddle point problems is due to the fact that the kernel space X_0^h is not a subset of X_0.

Lemma 4.3. ***(Coercivity of** a)* *Let $\alpha_c = \alpha_d = 1/2$. Then, with the above notation and definitions in place, there exists a constant $\gamma_a^h > 0$ such that:*

$$\inf_{U_h \in X_0^h} \sup_{V_h \in X_0^h} \frac{a\left(U_h, V_h\right)}{\|U_h\|_X \|V_h\|_X} > \gamma_a^h$$

with $\gamma_a^h = \gamma_a$.

Proof. The proof is actually similar to the one provided for Lemma 4.1. A sketch is provided here.

Observe that functions $V_h = (v_h, z)$ in X_0^h satisfy

$$b(\mu_h, V_h) = 0, \quad \forall \mu_h \in M^h$$

i.e.

$$\int_{\Omega_o} \beta_1 v_h \mu_h + \beta_2 v_h' \mu_h' dx = \int_{\Omega_o} \beta_1 (\Pi z) \mu_h + \beta_2 (\Pi z)' \mu_h' dx, \quad \forall \mu_h \in M^h$$

In other words, given a function $z \in V_d$, v_h is simply viewed as the projection of Πz on V_c^h if $M^h = V_c^h$. Take $\mu_h = 1$ on Ω_o; then, if $\beta_1 \neq 0$,

$$\int_{\Omega_o} v_h dx = \int_{\Omega_o} (\Pi z) dx, \quad \forall \mu_h \in M^h$$

The averages of v_h and Πz on Ω_o are equal but the functions are not necessarily identical unlike in the continuous case. However, if every particle on Ω_o coincides with a node of M^h (case (b) in Figure 4.6), then $v_h = \Pi z$. If not, only the equality of averages is necessary to show coercivity if $\beta_1 \neq 0$ – see Remark 4.1.

In the case where $\beta_1 = 0$, coercivity of the bilinear form is immediate. □

Remark 4.6. Coercivity of $a(\cdot, \cdot)$ is not established for the case where α_c and α_d are linear. The proof is of course straightforward when using the particle coupling and essentially follows the proof of Lemma 4.1 since $v_h = \Pi z$. However, in the general case, the proof becomes very technical as the elements of the space X_0^h become more complex.

Lemma 4.4. ***(Inf-Sup condition for*** b***)*** *With above notation and definitions, there exists a constant* $\gamma_b^h > 0$ *such that:*

$$\inf_{\mu_h \in M^h} \sup_{V_h \in X^h} \frac{b\left(\mu_h, V_h\right)}{\|\mu_h\|_M \|V_h\|_X} > \gamma_b^h$$

Proof. Let $\mu_h \in M^h$. Similarly to the continuous case, it is enough to show that

$$\sup_{V_h \in X^h} \frac{|b\left(\mu_h, V_h\right)|}{\|V_h\|_X} > \gamma_b^h \|\mu_h\|_M$$

with $\gamma_b^h > 0$ independent of μ_h. Consider the following two cases:

1. **Continuum/RVE coupling:** In this case, given $\mu_h \in M^h$, one can always find a function $\hat{v}_h \in V_c^h$ such that $\hat{v}_h = \mu_h - \mu_a$ on Ω_o and $\hat{v} = 0$ on $\Omega_c \backslash \Omega_o$, where $\mu_a = \mu^h(x_a)$. Furthermore, select $\hat{z} = \mu_a$ so that $\hat{V}_h = (\hat{v}_h, \hat{z})$. Thus,

   $$\sup_{V_h \in X^h} \frac{|b\left(\mu_h, V_h\right)|}{\|V_h\|_X} > \frac{\left|b\left(\mu_h, \hat{V}_h\right)\right|}{\|\hat{V}_h\|_X}$$

   The proof then follows the one in Lemma 4.2 and it is concluded that $\gamma_b^h = \gamma_b$.

2. **Particle coupling:** In this case, one can always find a vector $\hat{z} \in V_d$ such that $\Pi\hat{z} = \mu_h$ on Ω_o. On $\Omega_d \backslash \Omega_o$, $\hat{z}$ is chosen as a constant vector so that $\hat{V}_h = (0, \Pi\hat{z})$. Then:

   $$\sup_{V_h \in X^h} \frac{|b\left(\mu_h, V_h\right)|}{\|V_h\|_X} > \frac{\left|b\left(\mu_h, \hat{V}_h\right)\right|}{\|\hat{V}_h\|_X} = \frac{\|\mu_h\|_M^2}{\|\hat{z}\|_{V_d}}$$

   It remains to show that $\|\mu_h\|_M / \|\hat{z}\|_{V_d}$ is greater than a positive constant. Since $\hat{z}$ is constant on $\Omega_d \backslash \Omega_o$, then, using Lemma F.2,

   $$\begin{aligned}\|\hat{z}\|_{V_d}^2 &= \sum_{i=1}^{n_o} k_i (\hat{z}_i - \hat{z}_{i-1})^2 + \delta \left(\sum_{i=1}^{n_o} \frac{l_i}{|\Omega_o|} \left(\frac{\hat{z}_i + \hat{z}_{i-1}}{2} \right) \right)^2 \\ &\leq \max_i k_i l_i |\mu_h|_{H^1(\Omega_o)}^2 + \delta \bar{\mu}_h^2\end{aligned}$$

   If $\beta_1 = 0$, fix $\bar{\mu}_h = 0$ so that:

   $$\|\hat{z}\|_{V_d}^2 \leq \max_i k_i l_i |\mu_h|_{H^1(\Omega_o)}^2 = \frac{\max_i k_i l_i}{\beta_2} \|\mu_h\|_M^2$$

and $\gamma_b^h = \sqrt{\beta_2/\max_i k_i l_i}$.

If β_1 is non-zero, then using Lemma F.2,

$$\begin{aligned}
\|\hat{z}\|_{V_d}^2 &\leq \max_i k_i l_i |\mu_h|_{H^1(\Omega_o)}^2 + \frac{\delta}{|\Omega_o|}\|\mu_h\|_{L^2(\Omega_o)}^2 \\
&\leq \max\left(\frac{\delta}{\beta_1|\Omega_o|}, \frac{\max_i k_i l_i}{\beta_2}\right)(\beta_1\|\mu_h\|_{L^2(\Omega_o)}^2 + \beta_2|\mu_h|_{H^1(\Omega_o)}^2) \\
&= \max\left(\frac{\delta}{\beta_1|\Omega_o|}, \frac{\max_i k_i l_i}{\beta_2}\right)\|\mu_h\|_M^2
\end{aligned}$$

which completes the proof with:

$$\gamma_b^h = \min\left(\sqrt{\frac{\beta_1|\Omega_o|}{\delta}}, \sqrt{\frac{\beta_2}{\max_i k_i l_i}}\right)$$

□

Remark 4.7. Note that in the discrete case, the bilinear form $b(\cdot,\cdot)$ does satisfy the inf-sup condition if $\beta_2 = 0$. Indeed, in this case, the term $|\mu_h|_{H^1(\Omega_o)}$ can be bounded by $\|\mu_h\|_{L^2(\Omega_o)}$ using an inverse inequality. However, the inf-sup constant would be dependent on the mesh size h, and would go to zero as h tends to zero.

Remark 4.8. Note that, as pointed out by Ben Dhia [18], the discretization of the Lagrange multiplier space cannot be finer than the discretization of the continuum model *and* the particle spacing. This can be seen from the proof since one would not be able to find a v^h or Πz that is an extension of μ^h since it is possible $v^h \neq \mu^h$ in Ω_o.

Finally, the following theorem follows from the continuity on X_h and coercivity on X_0^h of $a(\cdot,\cdot)$, from the continuity of $l(\cdot)$ on X_h, and from the continuity and inf-sup condition of $b(\cdot,\cdot)$ on $M_h \times X_h$ (see Lemmas F.1, F.2, F.3, and Lemmas 4.3 and 4.4):

Theorem 4.2. *Problem (4.27) with $\beta_1 \geq 0$ and $\beta_2 > 0$ is well-posed, in the sense that the solution to (4.27) exists, is unique, and depends continuously on the data. Moreover, all constants are independent of h.*

a priori **Error Estimates**

For completeness, the following *a priori* error estimate is presented. The proof follows exactly that of the traditional mixed finite element error estimate (see e.g. [34]).

Theorem 4.3. *Let* $(u, w, \lambda) \in V_c \times V_d \times M$ *be the solutions to* (4.23) *and let* $(u_h, w_h, \lambda_h) \in V_c^h \times V_d \times M^h$ *be the solutions to* (4.27). *Then,*

$$\|(u - u_h, w - w_h)\|_X \leq C_1 \inf_{v_h \in V_c^h} \|u - v_h\|_{V_c} + C_2 \inf_{\mu_h \in M^h} \|\lambda - \mu_h\|_M$$

$$\|\lambda - \lambda_h\|_M \leq C_3 \inf_{v_h \in V_c^h} \|u - v_h\|_{V_c} + C_4 \inf_{\mu_h \in M^h} \|\lambda - \mu_h\|_M$$

where

$$C_1 = \left(1 + \frac{M_a}{\gamma_a^h}\right)\left(1 + \frac{M_b}{\gamma_b^h}\right), C_2 = \frac{M_b}{\gamma_a^h}$$

$$C_3 = \frac{M_a}{\gamma_b^h}\left(1 + \frac{M_a}{\gamma_a^h}\right)\left(1 + \frac{M_b}{\gamma_b^h}\right), C_4 = \left(1 + \frac{M_b}{\gamma_b^h} + \frac{M_a M_b}{\gamma_a^h \gamma_b^h}\right)$$

□

4.2.3 Numerical Examples

In all the following experiments, the domain $\Omega = (0, 3)$. Moreover, the force f_m applied at x_m is chosen in such a way that the displacement at the right end of the domain, when using the continuum model everywhere in Ω, is equal to unity. In what follows, the equilibrium lengths of the springs are all equal.

Uniform springs coefficients with α_c, α_d constant

In the first set of experiments, consider uniform springs such that $k = k_i = 1$, $i = 1, \ldots, m$. In this simple case, the solutions of the spring model and of the equivalent continuum model in all of Ω are linear. The continuum model is used in the subdomain $\Omega_c = (0, 2)$ while the particle model is used in $\Omega_d = (1, 3)$ and the weight coefficients α_c and α_d are chosen to be 1/2 in the overlap region. There are $m = 8$ springs in Ω_d, i.e. 9 particles. The equilibrium length of each spring is then given by $l = l_i = 0.25$. The continuum region is discretized with $N^e = 4$ elements. Because the springs are uniform, the representative cell used to derive the corresponding Young's modulus E is constituted of only one spring. Then

$$E = kl = 1 \times 0.25 = 0.25$$

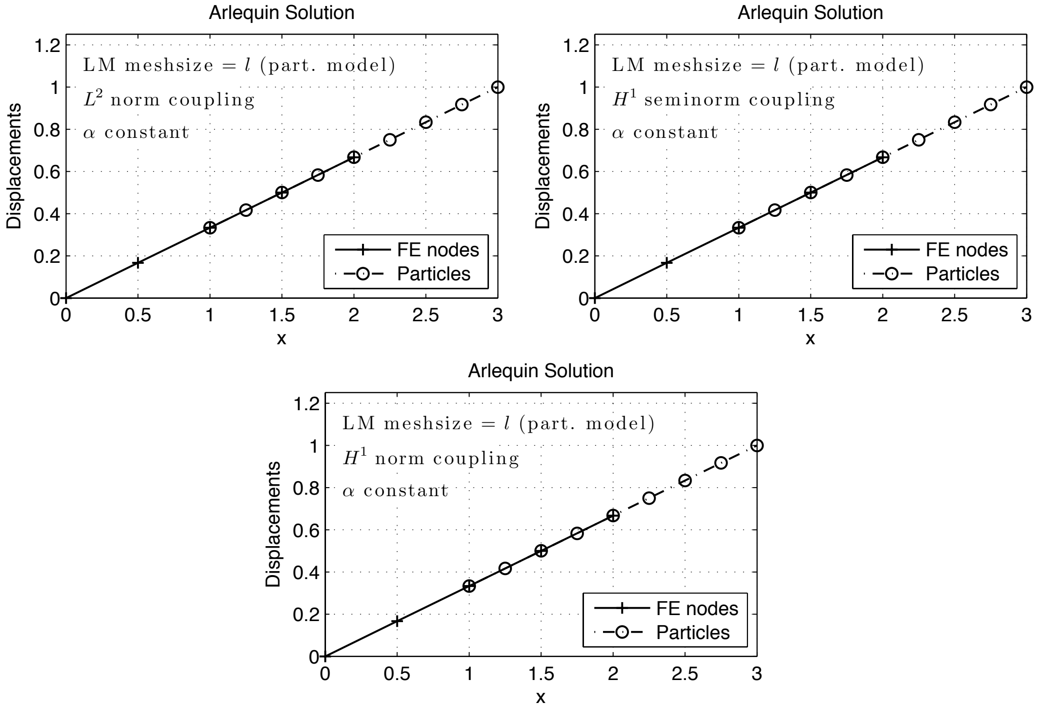


Figure 4.7: Uniform spring coefficients with particle coupling and α_c, α_d constant. The three graphs correspond to L^2 norm, H^1 seminorm, and H^1 norm coupling cases.

Consider first the case where the two models are coupled via a particle coupling, that is, the finite element space M^h for the Lagrange multipliers is dictated by the particles. As expected, this coupling ensures that the solutions of the Arlequin problem (4.27) are linear and that the continuum part exactly coincides with the particle solution over the overlap region in the three cases corresponding to the L^2 norm, H^1 seminorm, and H^1 norm couplings (see Figure 4.7). In these and subsequent plots, the initials LM refer to Lagrange multiplier. The displacement w_m at $x = 3$ is $w_m = 1$ in the three cases.

The experiment is repeated using continuum coupling, i.e. the elements in M^h are the same as in V_c^h on the overlap region. The coupling is therefore "weaker" than in the preceding experiment. The computed displacement at $x = 3$ is now $z_m = 1$ for the H^1 seminorm coupling, but $z_m = 1.01042$ in the other two cases (see Figure 4.8).

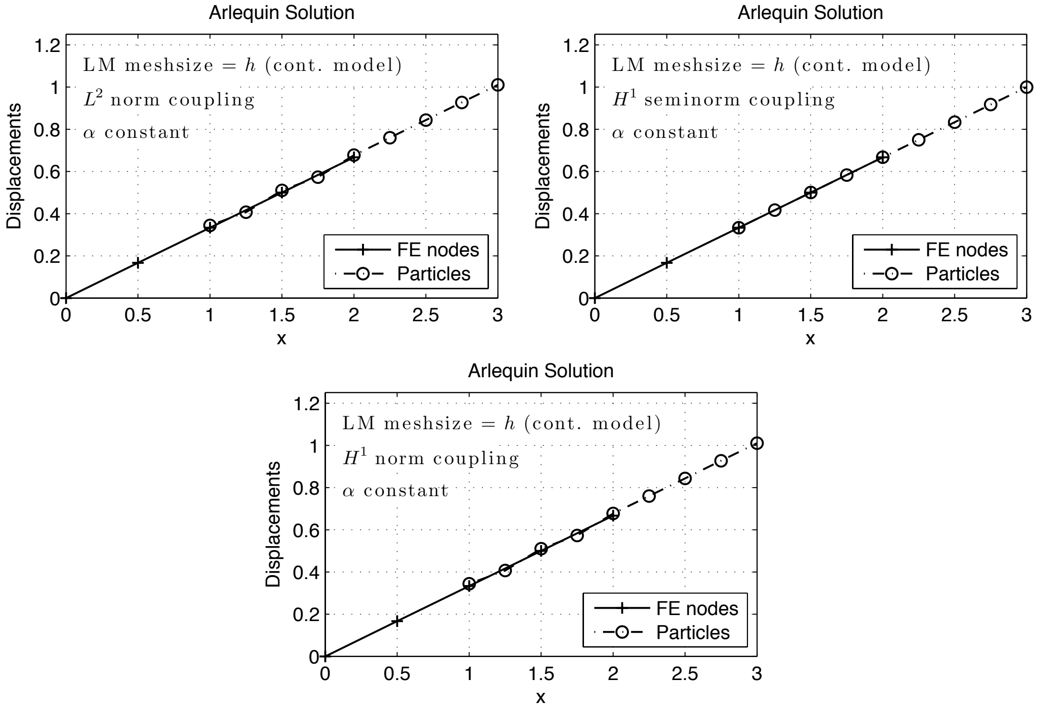


Figure 4.8: Uniform spring coefficients with continuum coupling and α_c, α_d constant.

Non-uniform stiffness coefficients with α_c, α_d constant

In more general settings, the interest is in problems in which the spring coefficients are not necessarily uniform but possibly randomly distributed, i.e. a polymer. As a simple test case, consider a periodic distribution of springs with two spring stiffness constants $k_1 = 100$ and $k_2 = 1$. For m even:

$$\begin{aligned} k_{2j-1} &= k_1 \quad j = 1, \ldots, m/2 \\ k_{2j} &= k_2 \quad j = 1, \ldots, m/2 \end{aligned}$$

As before, consider the following geometry and discretization data: $\Omega_c = (0, 2)$, $\Omega_d = (1, 3)$ $m = 8$, and $N^e = 4$. The equilibrium length of the springs is once again equal to $l = l_i = 0.25$. Using a representative cell (or Representative Volume Element, RVE) made of two consecutive springs, it follows that the Young's modulus is given by, :

$$E = \frac{k_1 k_2}{k_1 + k_2} 2l = \frac{100}{101} 0.5 = 0.49505$$

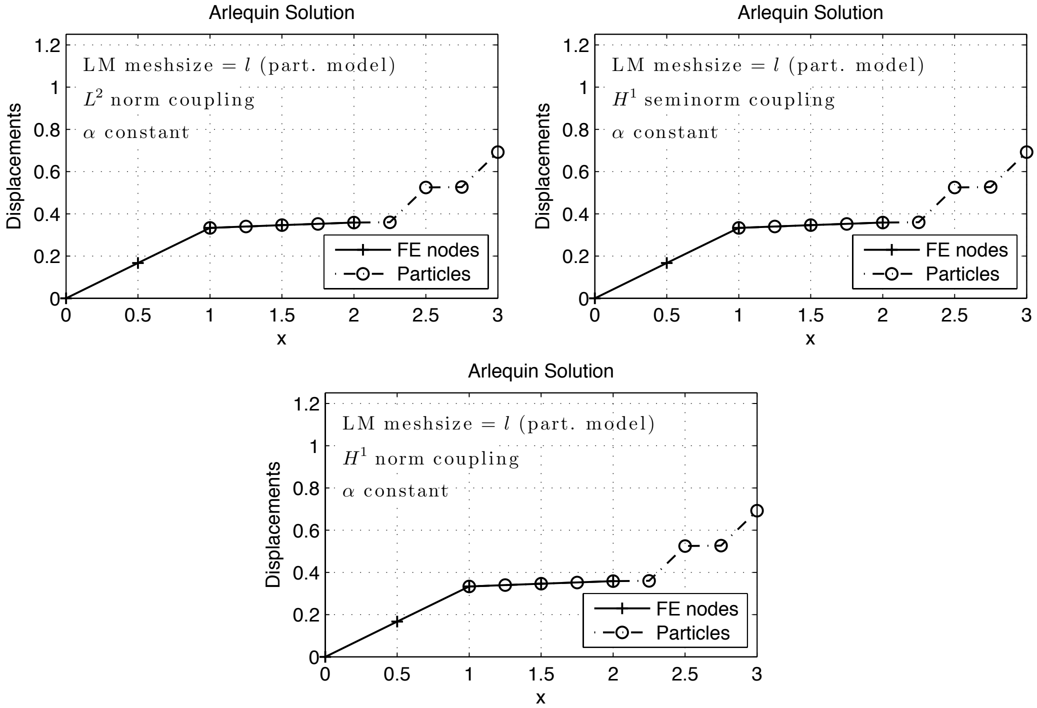


Figure 4.9: Periodic distribution of spring coefficients with particle coupling and α_c, α_d constant. The three graphs correspond to L^2 norm, H^1 seminorm, and H^1 norm coupling cases.

Figure 4.9 shows the Arlequin solutions in the case of particle coupling. It is not surprising that $z_m = 0.691822$ in the three cases of coupling since such a coupling is necessarily too constraining.

In this problem, it is clear that the elements in M_h should not be smaller than the representative cell used to derive the continuum model. For the continuum coupling, it is observed that the size of the elements in M_h is equal to the size of one representative cell, i.e. $h = 2l = 0.5$. Figure 4.10 shows the results when using continuum coupling. Observe that $z_m = 1$ for the H^1 seminorm coupling, but $z_m = 1.08727$ and $z_m = 1.08710$ for the L^2 and H^1 norm coupling, respectively. It is noted that in the H^1 seminorm case, the constant modes of V_d are fixed by setting z_0 to be equal to the displacement u_h at x_a.

Remark 4.9. A slight change in the slope of the continuum displacement u is observed in Figure 4.10. This variation can be interpreted by writing the equilibrium equation

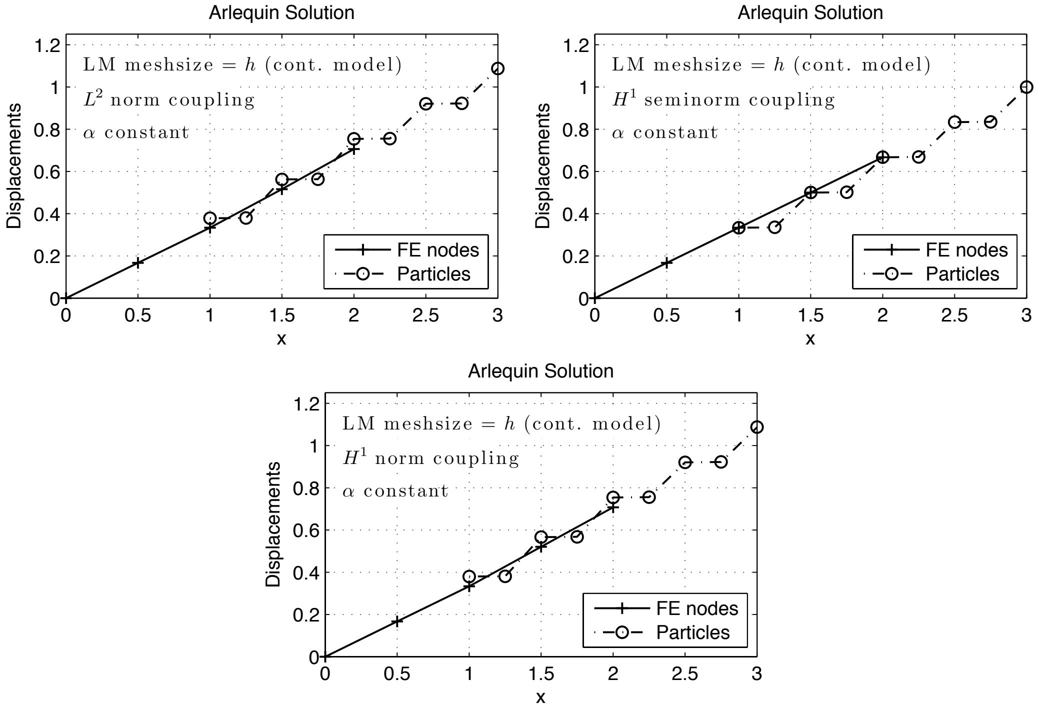


Figure 4.10: Periodic distribution of spring coefficients with continuum coupling and α_c, α_d constant.

at the interfacial point x_a. It is true that:

$$E\left.\frac{du}{dx}\right|_{x_a^-} = \alpha_c E\left.\frac{du}{dx}\right|_{x_a^+} + \alpha_d k_1 l_1 \frac{w_1 - w_0}{l_1} \tag{4.28}$$

Because $\alpha_d = 1/2$ here, and thus does not vanish at x_a, nothing guarantees that the two derivatives should be the same on the left and right sides of x_a. This issue is therefore inherent to the choice α_c and α_d constants and should be improved by the use of linear or cubic weight coefficients (see next subsection). Note this was also observed in [37].

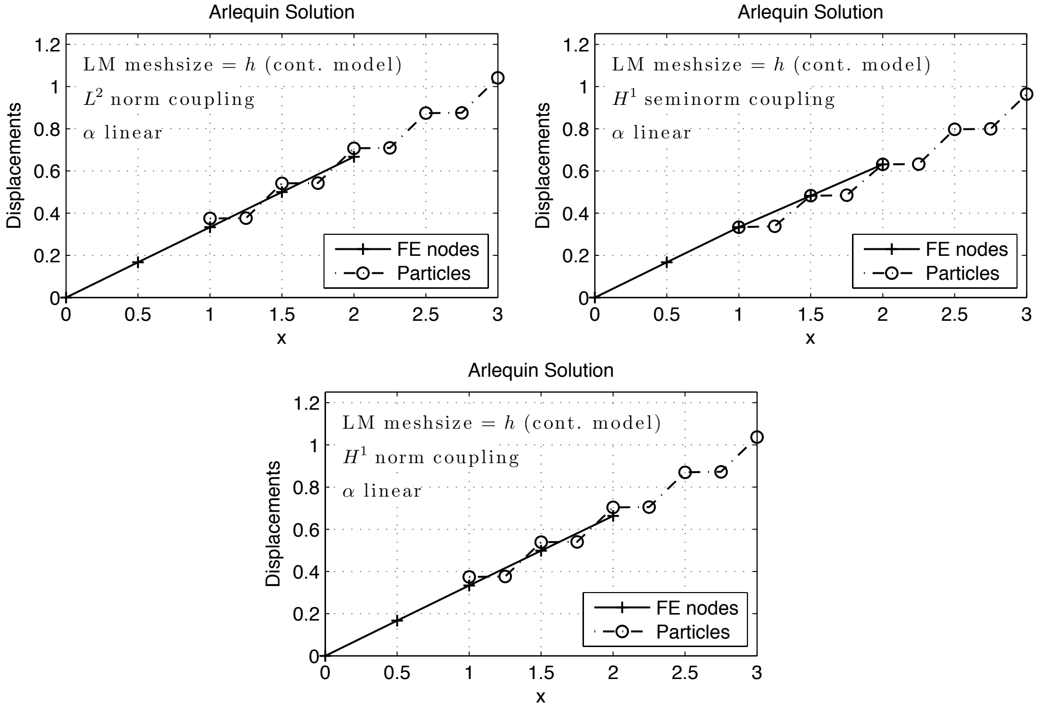
Influence of the weight coefficient α

In this subsection, the effect of using linear and cubic weight coefficients is studied. The same case is considered here as that studied in the previous subsection with continuum coupling. Figures 4.11 and 4.12 show the results with α linear and cubic, respectively. Observe that the change in slope in the continuum displacement u is no longer visible for the L^2 and H^1 norm couplings. However, a variation in the slope has appeared for the H^1 seminorm coupling. An explanation for this behavior is not available at this time.

The linear and cubic cases apparently provide similar results with a slight difference: the displacements of the particle at x_m with the L^2 and H^1 norm couplings are $z_m = 1.04084$ for the linear case and $z_m = 1.03707$ for the cubic case. These values are nevertheless greatly improved over the constant case for which approximately $z_m = 1.087$.

Representative volume element

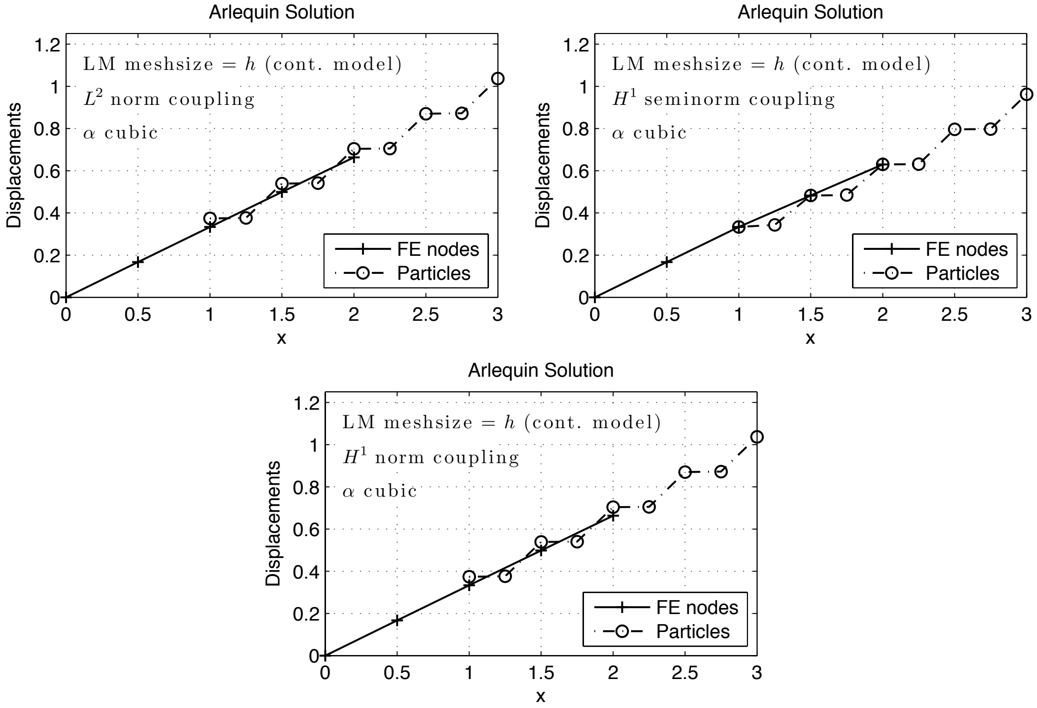
The objective in this subsection is to show that the mesh size h for the continuum solution can be chosen arbitrarily from the equilibrium length l of the particles, but that it is important to select the size of the elements for the Lagrange multiplier at least equal to the size, denoted ε, of the representative cell or volume element. Note that the continuum coupling case then becomes a subcase of this configuration. Figure 4.13 shows the results with α linear when $h = l/2$ and the meshsize for the Lagrange multiplier is equal to ε, which in this problem is simply $2l$. The results are exactly identical to the results obtained in Figure 4.11 for the L^2 and H^1 norm


Arlequin Solution

Arlequin Solution

Arlequin Solution

Figure 4.11: Same as Figure 4.10 but with α linear.


Arlequin Solution

Arlequin Solution

Arlequin Solution

Figure 4.12: Same as Figure 4.10 but with α cubic.

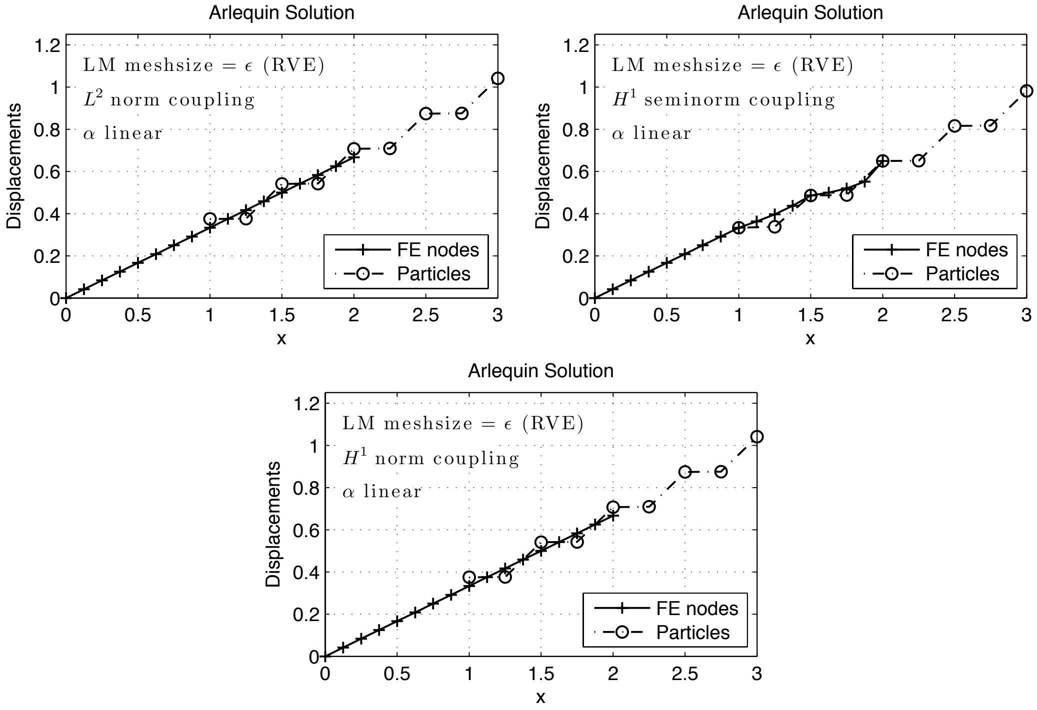


Figure 4.13: Same as Figure 4.11 but with $h = l/2$ and element size for the Lagrange multiplier (LM) equal to $2l = \varepsilon$.

couplings. However, the behavior of the continuum solution in the overlap region when using the H^1 seminorm coupling has the tendency to follow that of the particle solution. This is attributed to the fact that this type of coupling provides insufficient constraints on the two displacement fields. These results suggest that the H^1 seminorm coupling should not be retained as a useful candidate for these types of simulations.

Influence of mesh size

In this section, the effect of the mesh size on the Arlequin solution is studied. The equilibrium length of the springs the same as in Section 4.2.3 and the size of the elements in V_c^h is varied from $h = 1$ to $h = 1/32$. The stiffness of the springs is the same as in Section 4.2.3 and the continuum coupling is used here.

The displacements at $x = 3$ are collected in Table 4.6 for the different mesh sizes h and coupling types based on the L^2 norm, H^1 seminorm, and H^1 norm. Here, the weight coefficients α_c and α_d are chosen to be linear. For the L^2 and H^1 norms,

Table 4.6: Displacements z_m at $x = 3$ for various mesh sizes and coupling types. The equilibrium length of each spring is $l = 0.0625$.

h	L^2 norm	H^1 seminorm	H^1 norm
1	1.04084	0.994358	1.04084
1/2	1.04084	0.964384	1.04084
1/4 – 1/32	0.930203	0.930203	0.930203

the displacement at $x = 3$ are constant for every value of h until $h = 1/4$ and then the value remains constant again. This shows that the solution is exact for every $h \leq 1/4$ (i.e. the spacing of the particles), while for $h > 1/4$, the "average" solution is linear and is resolved exactly with linear elements. For the H^1 seminorm, the results improve as h decreases. Here, the solution is not exact due the constant chosen so, as the mesh is refined, the constraint becomes enforced more exactly, until $h \leq 1/4$ where the solution becomes exact.

The Arlequin solution and Lagrange multiplier are shown in Figures 4.2.3 and 4.14 , respectively, for $h = 1/2$. The same results for the case $h = 1/8$ are shown in Figures 4.15 and 4.16 and then for $h = 1/32$ in Figures 4.17 and 4.18. Note that the Lagrange multipliers are constant for the L^2 and H^1 norms cases, and smooth for the H^1 seminorm coupling when $h = 1/2$. For the L^2 norm, it is observed that the Lagrange multiplier μ_h displays larger and larger variations as the mesh is refined. This result is commensurate with the theoretical results in the sense that the discrete inf-sup constant goes to zero linearly with h if β_2 is set to zero. Note also how the linear α is reflected in the character of the Lagrange multiplier solution - at the interface of the overlap and discrete domains, the Lagrange multiplier solution is zero when $h \leq 1/4$.

Reconstruction of solutions

In the overlap region, the Arlequin method produces two solutions, one corresponding to the continuum model and the other to the particle model. Neither of these represents the solution of the problem at hand. It seems natural here to reconstruct a displacement field by combining the two solutions on the overlap region. This can be done in two ways. First, one can reconstruct a displacement field as follows:

$$\hat{u}(x) = \alpha_c u_h(x) + \alpha_d \Pi z(x), \qquad \forall x \in \Omega_o$$

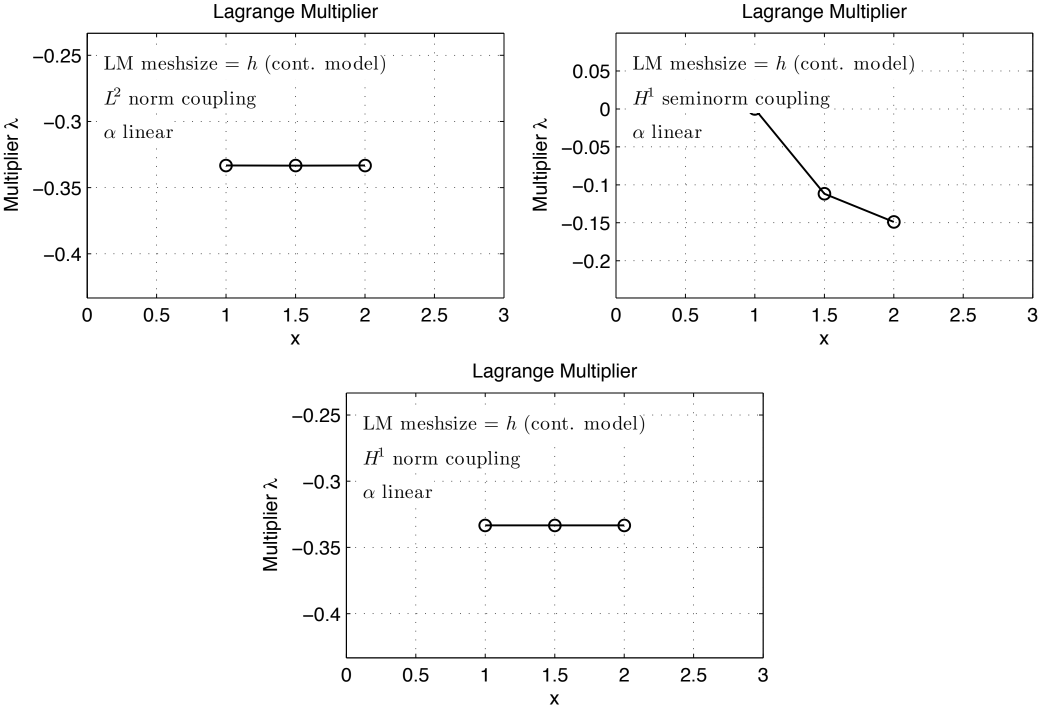


Figure 4.14: Lagrange multiplier solution in the case $l = 1/4$ and $h = 1/2$ using the continuum coupling and α linear.

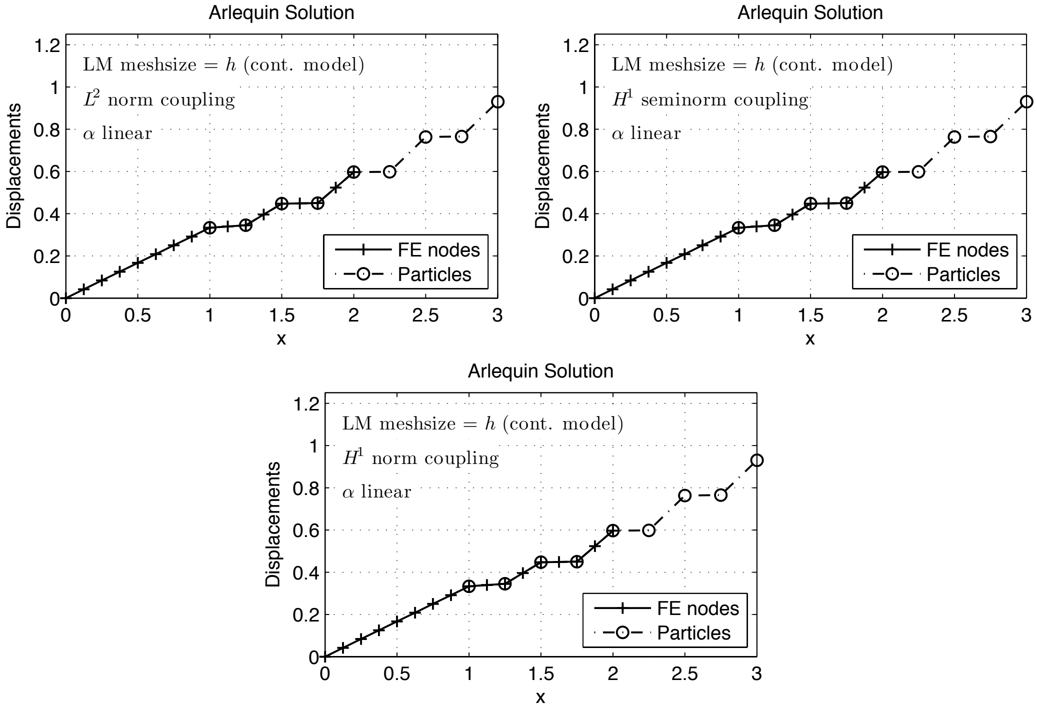


Figure 4.15: Arlequin solution in the case $l = 1/4$ and $h = 1/8$ using the continuum coupling and α linear.

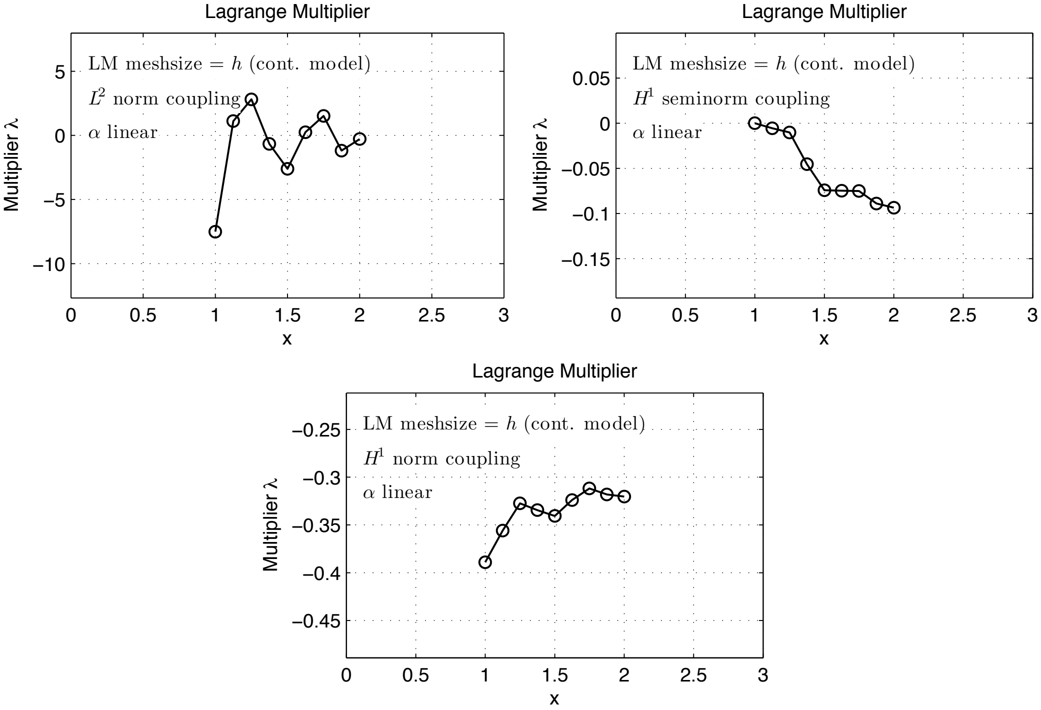


Figure 4.16: Lagrange multiplier solution in the case $l = 1/4$ and $h = 1/8$ using the continuum coupling and α linear.

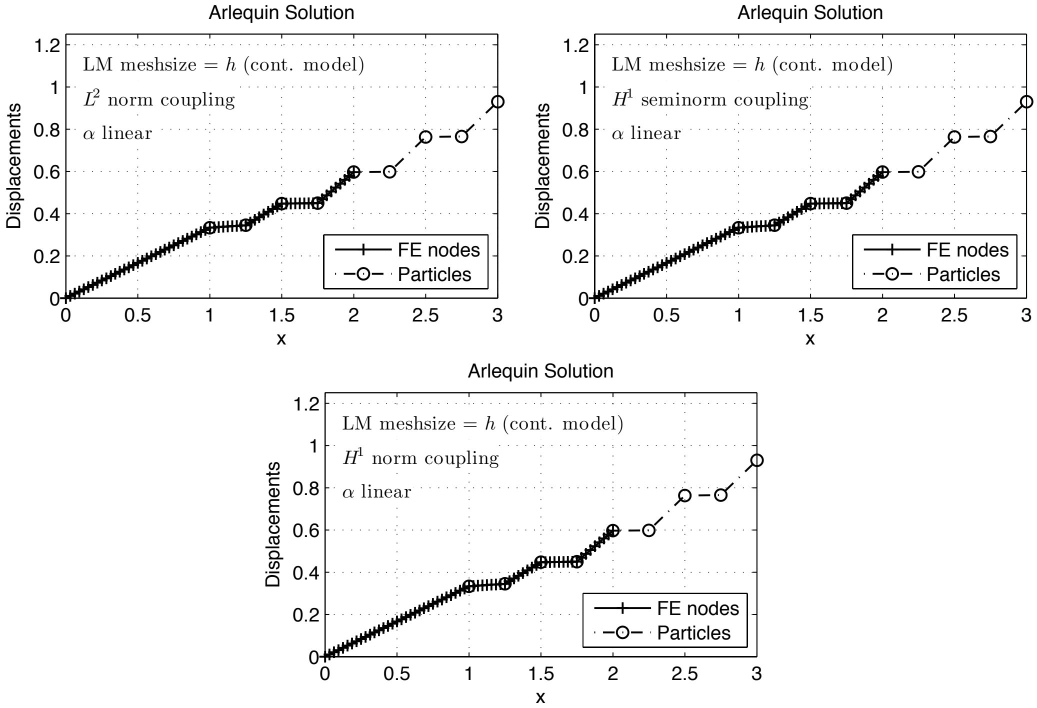


Figure 4.17: Arlequin solution in the case $l = 1/4$ and $h = 1/32$ using the continuum coupling and α linear.

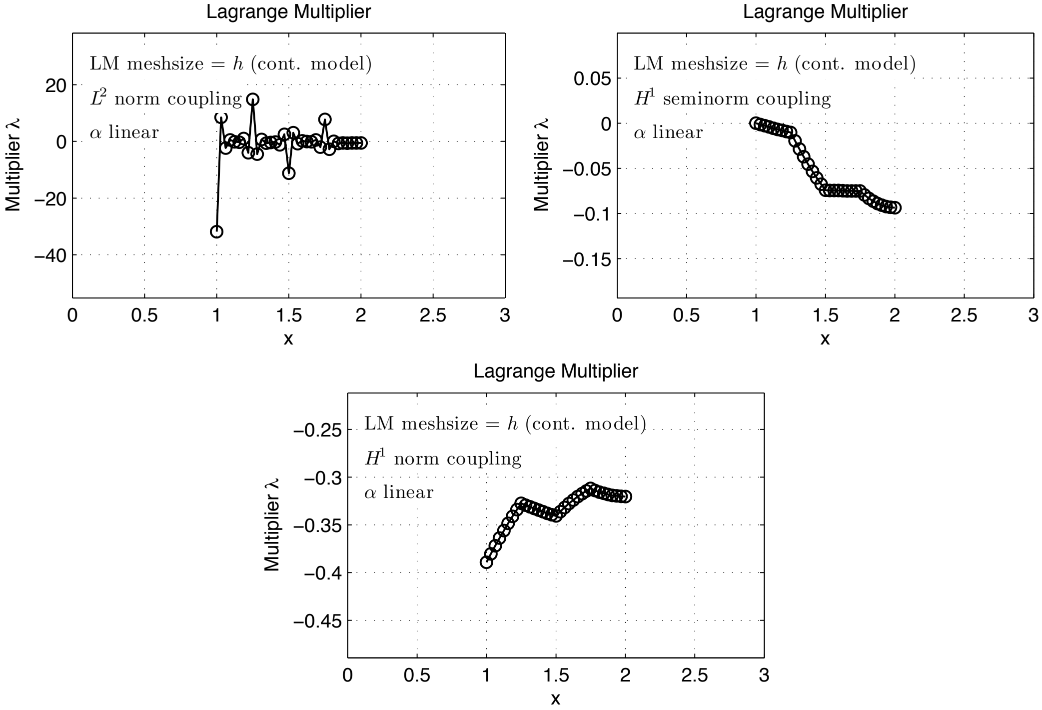


Figure 4.18: Lagrange multiplier solution in the case $l = 1/4$ and $h = 1/32$ using the continuum coupling and α linear.

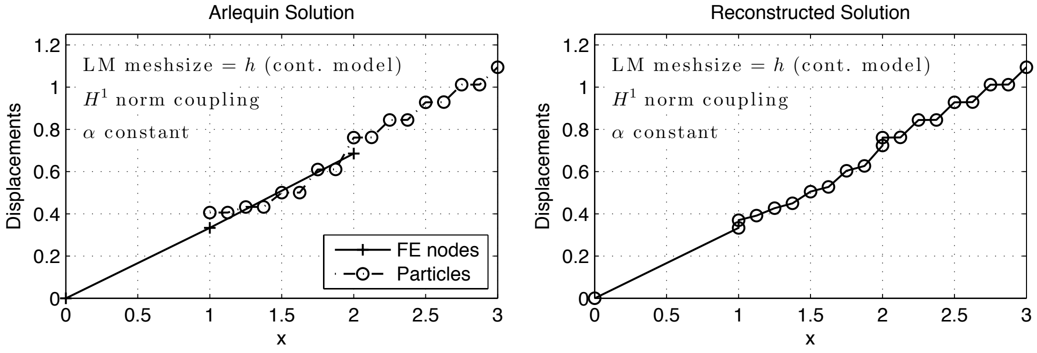


Figure 4.19: Arlequin solution and reconstructed solution using a continuum coupling for the Lagrange multiplier and the H^1 norm coupling with α constant.

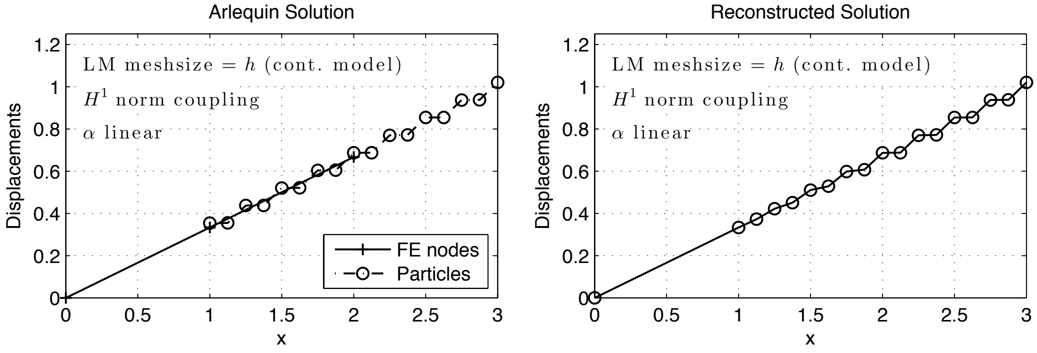


Figure 4.20: Arlequin solution and reconstructed solution using a continuum coupling for the Lagrange multiplier and the H^1 norm coupling with α linear.

Second, a displacement vector is reconstructed as:

$$\hat{z}_i = \alpha_c u_h(x_i) + \alpha_d z_i, \qquad \forall i = 1, \ldots, n_o$$

the Arlequin solution and the reconstructed solution are shown in Figure 4.19 for the case in which continuum coupling and H^1 norm coupling, along with constant weight coefficients α_c and α_d, are used. Here $N^e = 2$, and there are eight springs distributed over each element. It is observed that the reconstructed solution is discontinuous at both end points of the overlap domain and that the displacements display a relatively erratic behavior in Ω_o. The same results are shown in Figures 4.20 with linear weight coefficients and the respective solutions look much smoother.

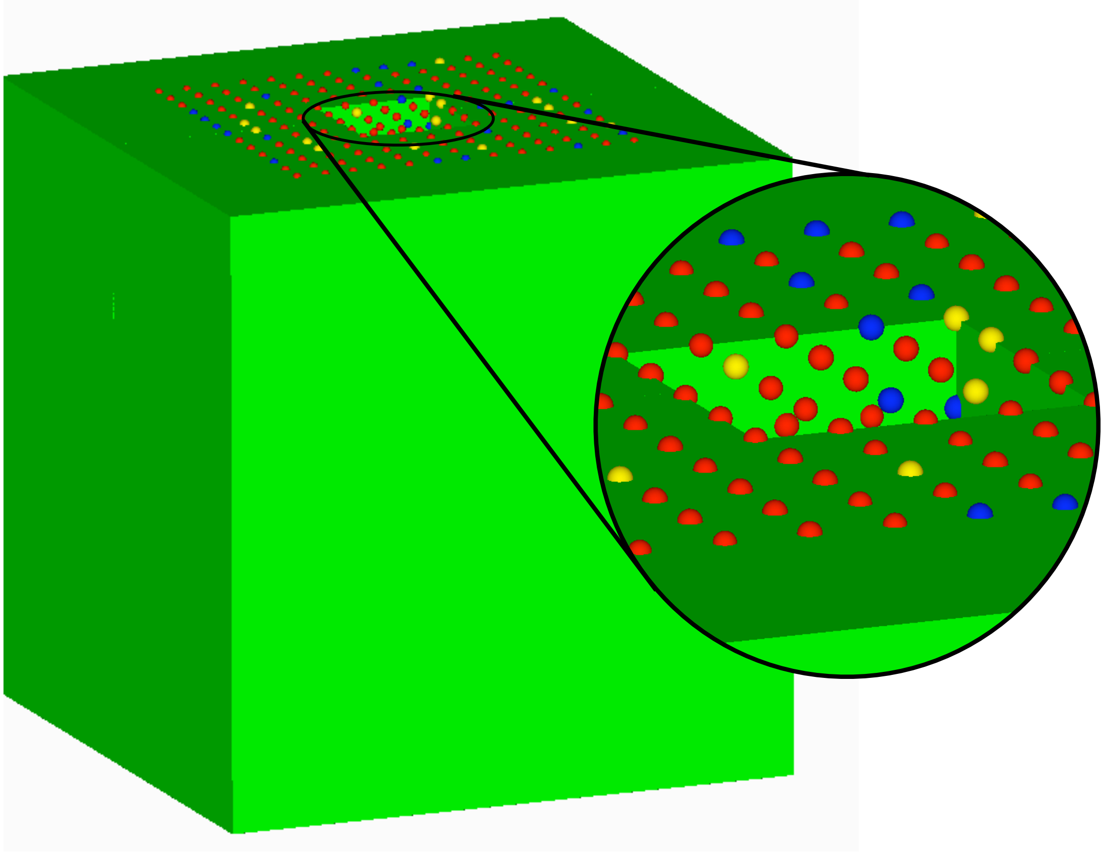


Figure 4.21: Example of a geometry for the arlequin method. The solid green represents the continuum model while the red, blue, and yellow particles represent monomer, cross-linker, and initiator, respectively. Notice that the zoomed shows the particle region, while the particles contained in the green define the overlap region.

4.3 3-D Arlequin Model of a Polymer

4.3.1 Continuous Formulation

Let $\Omega \subset \mathbb{R}^3$ be open and bounded. Let $\Omega_c \subset \Omega$ be the continuum region of the body and let $\Omega_d \subset \Omega$ be the region containing M discrete particles (representing the lattice polymer model in this context). Let $\Omega_o = \Omega_c \bigcap \Omega_d$ be the region of overlap. See

Figure 4.21 for example. Next, let

$$\begin{aligned} V_c &= \left\{ \mathbf{v} : \Omega_c \to \mathbb{R}^3 : \left| \int_{\Omega_c} W\left(\nabla \mathbf{u}\right) d\mathbf{x} \right| \neq \infty, \gamma_0 \mathbf{v} = 0 \text{ on } \Gamma_c^D \right\} \\ V_d &= \left\{ \mathbf{z} \in \mathbb{R}^{3M} : \mathbf{z} = 0 \text{ on } \Gamma_d^D \right\} \\ V_o &= \left\{ \mu \in H^1(\Omega_o)^3 \right\} \end{aligned} \tag{4.29}$$

Then, the Arlequin method proposes to solve the following saddle point problem:

$$\boxed{\inf_{(\mathbf{v},\mathbf{z}) \in V_c \times V_d} \sup_{\mu \in V_o} L(\mathbf{v}, \mathbf{z}, \mu)} \tag{4.30}$$

where

$$L(\mathbf{v}, \mathbf{z}, \mu) = E_c(\mathbf{v}) + E_d(\mathbf{z}) + b\left(\mu, \mathbf{v} - \Pi \mathbf{z}\right) \tag{4.31}$$

and

$$\begin{aligned} E_c(\mathbf{v}) &= \int_{\Omega_c} \alpha_c W\left(\nabla \mathbf{v})\right) d\mathbf{X} \\ E_d(\mathbf{z}) &= \frac{1}{2} \sum_{m=1}^{M} \sum_{n=1}^{n_m} \alpha_d E_{mn}\left(\|\mathbf{x}_m - \mathbf{x}_n\|\right) - \sum_{m=1}^{M} \mathbf{f} \cdot \left(\mathbf{x}_m - \mathbf{x}_m^0\right) \\ b\left(\mu, \mathbf{v} - \Pi \mathbf{z}\right) &= \int_{\Omega_o} \mu \cdot \left(\mathbf{v} - \Pi \mathbf{z}\right) + \nabla \mu : \nabla\left(\mathbf{v} - \Pi \mathbf{z}\right) d\mathbf{X} \\ \alpha_c, \alpha_d &\in \mathbb{R} : \alpha_c + \alpha_d = 1 \text{ in } \Omega \\ \alpha_c &= \begin{cases} 1 & \text{in } \Omega_c / \Omega_o \\ 0 & \text{in } \Omega_d \end{cases} \\ \alpha_d &= \begin{cases} 1 & \text{in } \Omega_d / \Omega_o \\ 0 & \text{in } \Omega_c \end{cases} \\ \Pi &: V_d \to V_c \end{aligned} \tag{4.32}$$

where, as in the one-dimensional case, Π is the linear interpolant.

4.3.2 Discrete Formulation

Let $V_c^h \subset V_c$ and $V_o^h \subset V_o$ be discretizations of the continuum and lagrange multiplier function spaces. See Figure 4.22 for example. In all examples that follow, the La-

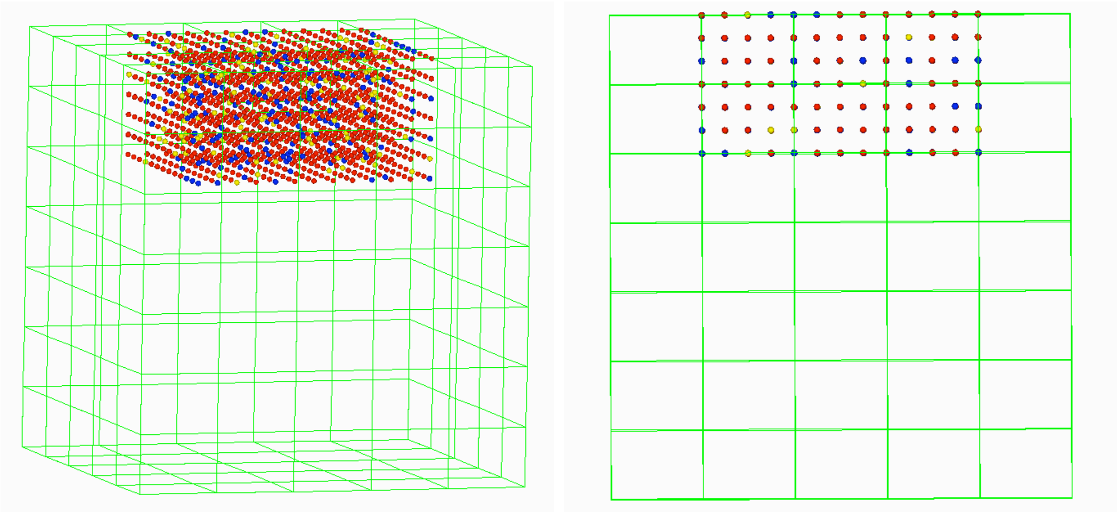


Figure 4.22: The discretization of the geometry in Figure 4.21 from an angle view and a side view.

grange multiplier finite elements will correspond with the continuum finite elements in the overlap region. The discrete formulation is as follows:

$$\boxed{\inf_{\left(\mathbf{v}^h,\mathbf{z}\right)\in V_c^h\times V_d}\ \sup_{\mu^h\in V_o^h} L(\mathbf{v}^h,\mathbf{z},\mu^h)} \tag{4.33}$$

where, as before,

$$L(\mathbf{v}^h,\mathbf{z},\mu^h) = E_c(\mathbf{v}^h) + E_d(\mathbf{z}) + b\left(\mu^h, \mathbf{v}^h - \Pi\mathbf{z}\right) \tag{4.34}$$

Let N_A^i represent the finite element basis functions for the continuum model with A being the global node number and i the component of the vector (i.e. x, y, z), N_B^i the basis for the lagrange multipliers, and N_m^i the basis functions for interpolating the particles in Ω_o. Then,

$$u_i^h = \sum_{A=1}^{N_c} d_{\underline{i}A} N_A^{\underline{i}}, \quad \lambda_i^h = \sum_{B=1}^{N_l} l_{\underline{i}B} N_B^{\underline{i}}, \quad \Pi^h x_i = \sum_{m=1}^{N_I} x_{\underline{i}m} N_m^{\underline{i}} \tag{4.35}$$

Thus, the Lagrangian can be rewritten as

$$
\begin{aligned}
L(d_{iA}, x_{im}, l_{iB}) = &\int_{\Omega_c} \alpha_c W \left(\nabla \left(\sum_{A=1}^{N_c} d_{\underline{i}A} N_A^{\underline{i}} \right) \right) d\mathbf{X} - \\
&\int_{\Gamma_c^N} T_i \left(\sum_{A=1}^{N_c} d_{\underline{i}A} N_A^{\underline{i}} \right) d\Gamma_N + \\
&\frac{1}{2} \sum_{m=1}^{M} \sum_{n=1}^{N_m} \alpha_d E_{mn} \left(\| x_{im} - x_{in} \| \right) - \sum_{m=1}^{M} f_{im} \left(x_{im} - x_{im}^0 \right) + \\
&\int_{\Omega_o} \left(\sum_{B=1}^{N_l} l_{\underline{i}B} N_B^{\underline{i}} \right) \left(\sum_{A=1}^{N_c} d_{\underline{i}A} N_A^{\underline{i}} - \sum_{m=1}^{N_I} x_{\underline{i}m} N_m^{\underline{i}} \right) d\mathbf{X}
\end{aligned} \tag{4.36}
$$

A necessary condition for a saddle point is that the derivative be zero. In particular, set

$$
\frac{\partial L}{\partial d_{jD}} = 0, \quad \frac{\partial L}{\partial x_{im}} = 0, \quad \frac{\partial L}{\partial l_{jE}} = 0 \tag{4.37}
$$

Thus, using relationships and procedures from Appendix E, (4.37) yields the following system of nonlinear equations:

$$
\boxed{
\begin{aligned}
&\int_{\Omega_c} \alpha_c S_{IJ} F_{\underline{i}I} N_{D,J}^{\underline{i}} d\mathbf{X} + \int_{\Omega_o} \left(\sum_{B=1}^{N_l} l_{\underline{j}B} N_{B,I}^{\underline{j}} \right) N_{D,I}^{j} d\mathbf{X} = \int_{\Gamma_c^N} T_{\underline{j}} N_D^{\underline{j}} d\Gamma_N \\
&\sum_{n=1}^{N_m} \alpha_d E'_{nm} \left(\| x_{\underline{im}} - x_{\underline{in}} \| \right) \frac{(x_{\underline{im}} - x_{\underline{in}})}{\| x_{\underline{im}} - x_{\underline{in}} \|} - \int_{\Omega_o} \left(\sum_{B=1}^{N_l} l_{\underline{j}B} N_{B,I}^{\underline{j}} \right) N_{m,I}^{\underline{i}} d\mathbf{X} = f_{im} \\
&\int_{\Omega_o} N_{E,I}^{\underline{j}} \left(\sum_{A=1}^{N_c} d_{\underline{i}A} N_A^{\underline{i}} - \sum_{m=1}^{N_I} x_{\underline{i}m} N_m^{\underline{i}} \right) d\mathbf{X} = 0
\end{aligned}
} \tag{4.38}
$$

Note that the initial strain can easily be incorporated into the continuum as in Section 3.5.

The nonlinear equations Newton with line search algorithm described in Section 2.4 is used to solve the above system. Repeating the differentiation procedure to arrive at (4.38), then the Jacobian is

$$
\mathbf{J} = \begin{bmatrix} J_c & 0 & J_{\lambda c} \\ 0 & J_d & J_{\lambda d} \\ J_{\lambda c}^T & J_{\lambda d}^T & 0 \end{bmatrix} \tag{4.39}
$$

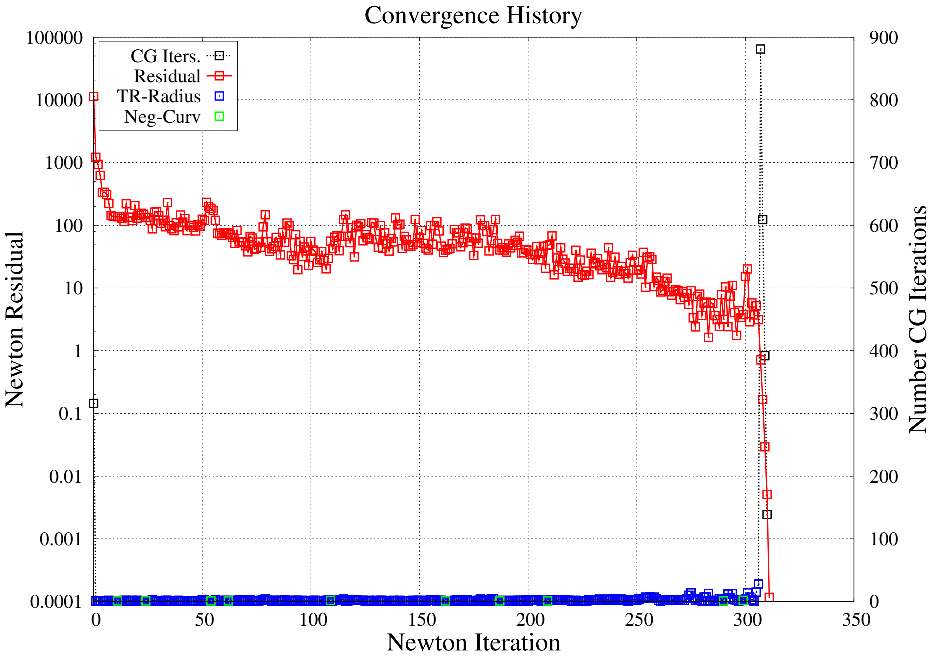
where

$$
\begin{aligned}
J_c &= \frac{\partial^2 L}{\partial d_{jD} \partial d_{kF}} = \int_{\Omega_c} \alpha_c N^{\underline{j}}_{F,I} S_{IJ} N^{\underline{j}}_{D,J} + F_{\underline{j}I} C_{IJKL} F_{\underline{k}K} N^{\underline{j}}_{D,J} N^{\underline{k}}_{F,L} d\mathbf{X} \\
J_{\lambda c} &= \frac{\partial^2 L}{\partial d_{jD} \partial l_{kG}} = \int_{\Omega_o} N^{\underline{k}}_{G,I} N^{\underline{k}}_{D,I} d\mathbf{X} \\
J_d &= \frac{\partial^2 L}{\partial x_{im} \partial x_{jn}} = \sum_{m=1}^{N_m} \sum_{n=1}^{N_m} \alpha_d \mathbf{H}_{mn} \\
J_{\lambda d} &= \frac{\partial^2 L}{\partial x_{im} \partial d_{kG}} = -\int_{\Omega_o} N^{\underline{k}}_{G,I} N^{\underline{k}}_{m,I} d\mathbf{X}
\end{aligned}
\qquad (4.40)
$$

and $\mathbf{H}_{mn}$ correspond to intermolecular Hessians such as those in Appendix C.

4.4 Numerical Example

For the following numerical example, it is assumed that $\alpha_c = \alpha_d = 1/2$, that the polymer model used corresponds to that introduced in Chapter 2, and that the continuum model is the nonlinear elastic Mooney-Rivlin material with parameters given in Table 3.3.2. Only the H^1 coupling is considered here. Consider the approximation of a $50 \times 50 \times 50$ polymer lattice by an Arlequin surrogate consisting of 139 trilinear finite elements with 17 more for Lagrange multiplers and $21 \times 21 \times 10$ particles. This corresponds to $18,228$ unknowns, compared to the $375,000$ unknowns for the full polymer model: a factor of almost 21 in savings. The shrinkage parameter $\lambda = 1.3$ for this case. The convergence results for both the base model and the surrogate are shown in Figure 4.23 and the equilibrium configuration of the surrogate is in Figure 4.24. The total solve time was approximately 1.4 CPU hours on 32 processors for the base model while the surrogate model converged in under 5 minutes on a single processor, a factor of roughly 17. Aside from the system being much smaller, the continuum part of the coupled model appears to smoothen the residual function as only 8 Newton iterations were required to attain convergence while over 300 were required for the base model. This represents a substantial performance gain.



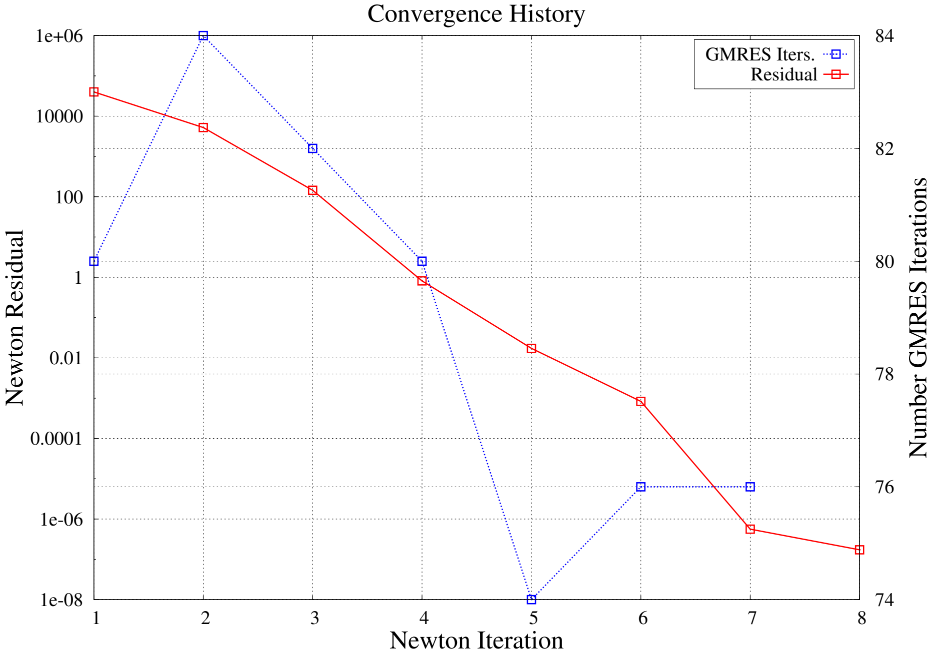


Figure 4.23: Convergence behavior of a 50 × 50 × 50 and its approximation by an Arlequin model coupling the particle with a nonlinear elastic continuum. The red curve shows the value of the residual and the blue curve is this number of C-G (GMRES) iterations required at each step.

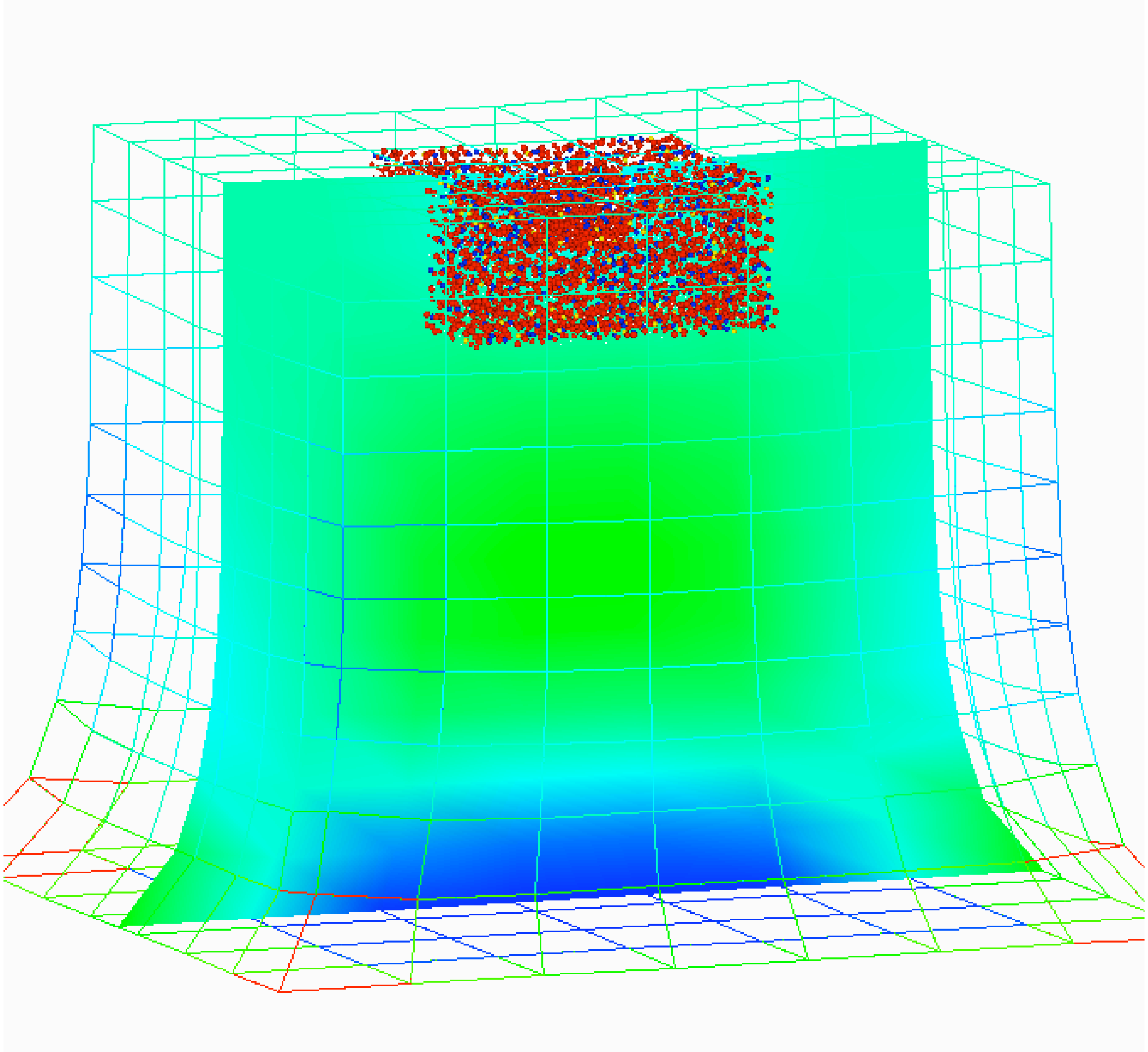


Figure 4.24: Equilibrium configuration of an Arlequin approximation to a $51 \times 51 \times 51$ polymer lattice. The mesh is colored by elements of the zz-component of the Cauchy stress while the slice is the interpolated zz-Cauchy stress. Red particles correspond to monomers, blue to crosslinkers, and yellow to initiators.

Chapter 5

Goal-Oriented Error Estimation and Adaptivity

5.1 Introduction

In this chapter, goal-oriented error estimation and adaptivity with the Goals algorithm are discussed in detail. In particular, the formulation of the adjoint problem introduced in Section 1.4 is posed for the full lattice problem (5.2). In addition, because the surrogate solution discussed in the previous chapter does correspond to a coupled particle and continuum model, schemes for projecting the surrogate solution to the lattice are discussed. With the adjoint solution, the residual can be computed as in Section 1.7 and an adaptive algorithm constructed. Such an algorithm is discussed in Section 5.4. Finally, numerical examples are shown in order to study the fidelity of the error estimates and the robustness of the adaptive algorithm.

5.2 Adjoint Problem and Calculation of Residual

From Section 1.2.1, the adjoint problem is: given a solution $\mathbf{u} \in U$ of (1.1) with $B(\cdot;\cdot)$ given by $(1.3)_3$, find $\mathbf{p} \in V$ such that

$$B'(\mathbf{u};\mathbf{v},\mathbf{p}) = Q'(\mathbf{u};\mathbf{v}) \qquad \forall \mathbf{v} \in V \tag{5.1}$$

where

$$B'(\mathbf{u};\mathbf{v},\mathbf{p}) = \sum_{i=1}^{N_0} \mathbf{v}_i \cdot \frac{\partial^2 E_i(\mathbf{u})}{\partial^2 \mathbf{u}} \cdot \mathbf{p}_i \tag{5.2}$$

However, in this case, the *exact* solution $\mathbf{u}$ is unknown; only the surrogate solution $\mathbf{u}_0$ is known. The surrogate $\mathbf{u}_0$ here corresponds to the solution of an Arlequin problem of the type in (4.38). Thus, the primal solution exists on the space X (the displacement of particles and continuum) and M (the Lagrange multipliers). Therefore, a projection $\Pi : X \to V$ must be used. In this case, the following map is used:

1. Particle part of Arlequin model: assign the value of the displacement to the corresponding particle in the base model
2. Continuum part of Arlequin model:
    - Loop over all particles contained in the continuum region
    - Determine element location of particle in finite element mesh reference configuration
    - Obtain master element coordinates (ξ, η, ζ) by solving $\mathbf{x}_i - \mathbf{x}_0(\xi, \eta, \zeta) = 0$ in the reference configuration
    - Evaluate $\mathbf{x}_0(\xi, \eta, \zeta)$ in the current configuration and assign the value to the particle in the lattice.
3. Overlap part of Arlequin model: there are several choices, but the combination $\alpha_c \mathbf{u} + \alpha_d \mathbf{x}$ is taken here.

Figure 5.1 illustrates the results of the above projection from an Arlequin approximation of a uniform lattice. Finally, the Lagrange multiplier is superfluous in the context of this projection and is neglected.

Thus, the following approximation to the adjoint problem is made:

$$\begin{aligned} &\text{Find } \hat{\mathbf{p}} \in V \text{ such that} \\ &B'(\Pi\mathbf{u}_0;\mathbf{v},\hat{\mathbf{p}}) = Q'(\Pi\mathbf{u}_0;\mathbf{v}) \qquad \forall \mathbf{v} \in V \end{aligned} \tag{5.3}$$

where $\hat{\mathbf{p}}$ is the adjoint solution corresponding to the projected surrogate solution. This notation is introduced in order to differentiate this approximate dual solution

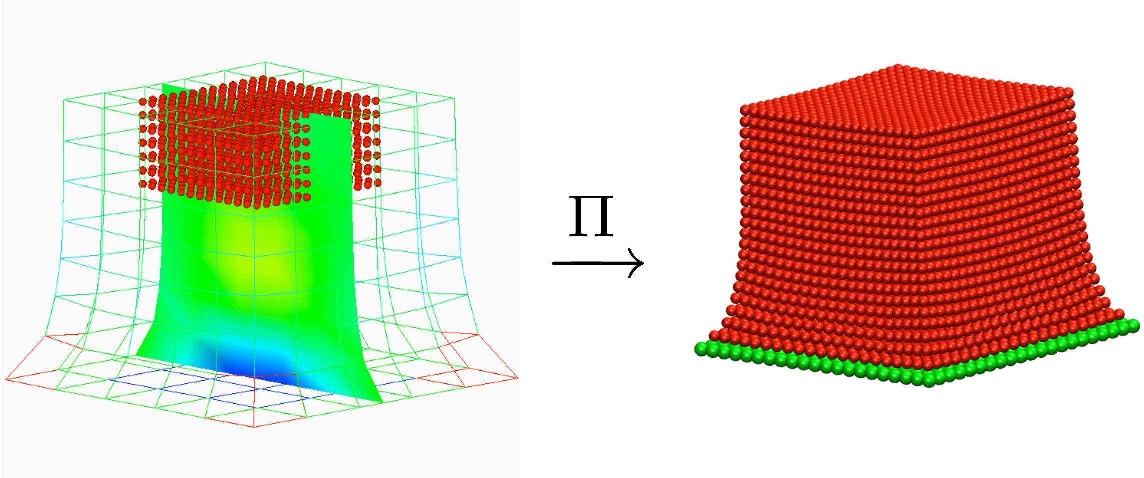


Figure 5.1: The result of projection Π on an Arlequin approximation of a $21 \times 21 \times 21$ uniform lattice. In the overlap region, the particle position is used for projecting the displacement, as opposed to the continuum.

from the exact dual solution $\mathbf{p}$. The only approximation made here is the use of $\Pi \mathbf{u}_0$ in place of $\mathbf{u}$.

With the calculation of the adjoint solution, the residual can be computed and errors in quantities of interest can be estimated. Recall that the error estimate is

$$\begin{aligned} \mathcal{E} &= Q(\mathbf{u}) - Q(\mathbf{u}_0) \\ &= \mathcal{R}(\mathbf{u}_0; \mathbf{p}) + \Delta \\ &= \mathcal{R}(\mathbf{u}_0; \mathbf{p}_0) + \mathcal{R}(\mathbf{u}_0; \mathbf{p} - \mathbf{p}_0) + \Delta \end{aligned} \tag{5.4}$$

In this case,

$$\begin{aligned} \mathcal{E} &= Q(\mathbf{u}) - Q(\Pi \mathbf{u}_0) \\ &= \mathcal{R}(\Pi \mathbf{u}_0; \mathbf{p}) + \Delta \\ &= \mathcal{R}(\Pi \mathbf{u}_0; \hat{\mathbf{p}}) + \mathcal{R}(\Pi \mathbf{u}_0; \mathbf{p} - \hat{\mathbf{p}}) + \Delta \end{aligned} \tag{5.5}$$

The Δ term is neglected in all examples that follow. The effect of neglecting the term $\mathcal{R}(\mathbf{u}_0; \mathbf{p} - \hat{\mathbf{p}})$ is investigated in the numerical examples.

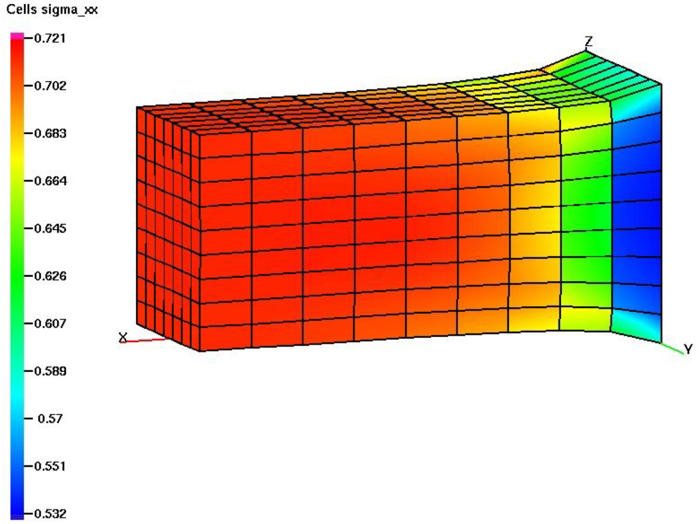


Figure 5.2: A 10^3 discretization of a cube. A uniform axial load is applied on the external (free) x-face while the zero x-face is fixed in all directions. A slice is taken to show the interior stress distribution.

5.3 Numerical Examples: Error Estimation

Polymer Lattice, Continuum Surrogate Preliminary results are given here to illustrate the solution of the finite element problem and doing preliminary error estimation calculations. Figure 5.2 shows a block under uniaxial loading in the positive x-plane and fixed in the zero x-plane.

Corresponding loading and boundary conditions are applied to a $20 \times 20 \times 20$ realization of the polymer lattice model (post relaxation). As an example, the quantity of interest is selected to be the length of the body. In particular, the differences in the average of the x-position of the positive and zero x-faces is taken as the quantity of interest:

$$Q(\mathbf{x}) = \frac{1}{N_{x+}} \sum_{i=1}^{N_{x+}} x_i^1 - \frac{1}{N_{x-}} \sum_{i=1}^{N_{x-}} x_i^1 \tag{5.6}$$

Thus, the right hand side of the adjoint problem (1.4) is

$$Q'(\mathbf{x}) = \begin{cases} \frac{1}{N_{x+}} & \text{if particle on } x+ \text{ face} \\ -\frac{1}{N_{x-}} & \text{if particle on } x- \text{ face} \\ 0 & \text{otherwise} \end{cases} \tag{5.7}$$

There is no approximation here to the adjoint solution so that the error in the quantity of interest is given by $(1.7)_2$.

This case does not represent a real physical system, but rather a model problem to perform a preliminary error analysis. Therefore, no units will be reported on the physical quantities that follow. Results of this test of the model error estimation are as follows: exact quantity of interest: $Q(\mathbf{x}) = 78.41$; surrogate approximation: $Q(\mathbf{x}_0) = 74.65$; effectivity index $\eta_{eff} = \frac{|Q(\mathbf{x})-Q(\mathbf{x}_0)|}{|\mathcal{R}(\mathbf{x}_0,\mathbf{p})|} = 1.05$. The effectivity index of 1.05 establishes an error of only five percent in the estimated error in the quantity of interest, a quite acceptable estimate for this class of problems. This result also suggests that for problems of this type, a continuum model may well be adequate for calculating such global quantities of interest.

Uniform Lattice, Arlequin Surrogate In this case, a 21 × 21 × 21 uniform lattice is taken and approximated by an Arlequin surrogate model. The constants for the springs in the base model are those given in uniform lattice examples in Section 2.5. The Arlequin surrogate model consists of 174 trilinear elements, 17 trilinear Lagrange multiplier elements (for the overlap region), and 1183 particles. The constitutive equations for the continuum is the compressible Mooney-Rivlin material with coefficients $C_1 = 1.0, C_2 = 1.0, C_3 = 1.0$. The shrinkage parameter $\lambda = 1.33$. The quantity of interest is the "slump": the average z-displacement of a 3 × 3 patch on the positive z-face of the body. That is,

$$Q(\mathbf{u}_0) = \frac{1}{|\mathcal{P}|} \sum_{i \in \mathcal{P}} u_i^z \tag{5.8}$$

where $\mathcal{P}$ is the set of particles in the 3 × 3 patch and u_i^z is the z-component of the displacement of particle i. Figure 5.3 displays the adjoint solution and Figure 5.4 the residual. Table 5.1 summarizes the results from the calculation.

In this example, the remainder $|\Delta|$ is seen to be approximately 8% of the total error in the quantity of interest.

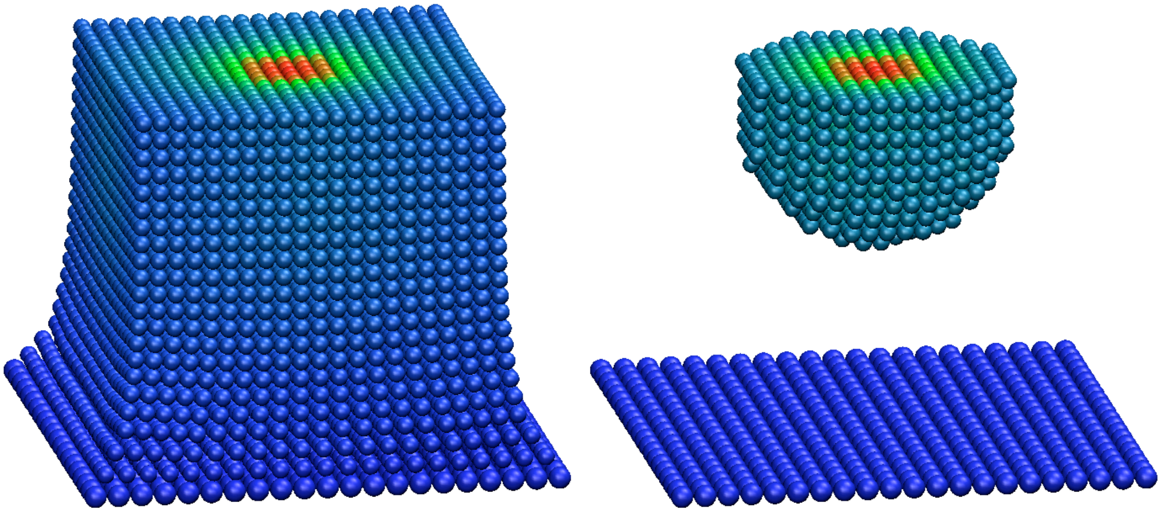


Figure 5.3: **(Left)** the adjoint solution $\hat{p}$ corresponding to an Arlequin approximation of a $21 \times 21 \times 21$ uniform lattice. The position of the particles is in the equilibrated current configuration. The value $\|\hat{\mathbf{p}}_i\| / \max_i \hat{\mathbf{p}}_i$ for each particle i is assigned a color in the figure, with red the highest values and blue the lowest. **(Right)** Here, the substrate particles are shown along with particles that have $\|\hat{\mathbf{p}}_i\| / \max_i \hat{\mathbf{p}}_i > 0.3$. This represents the region that most strongly influences the value of the quantity of interest.

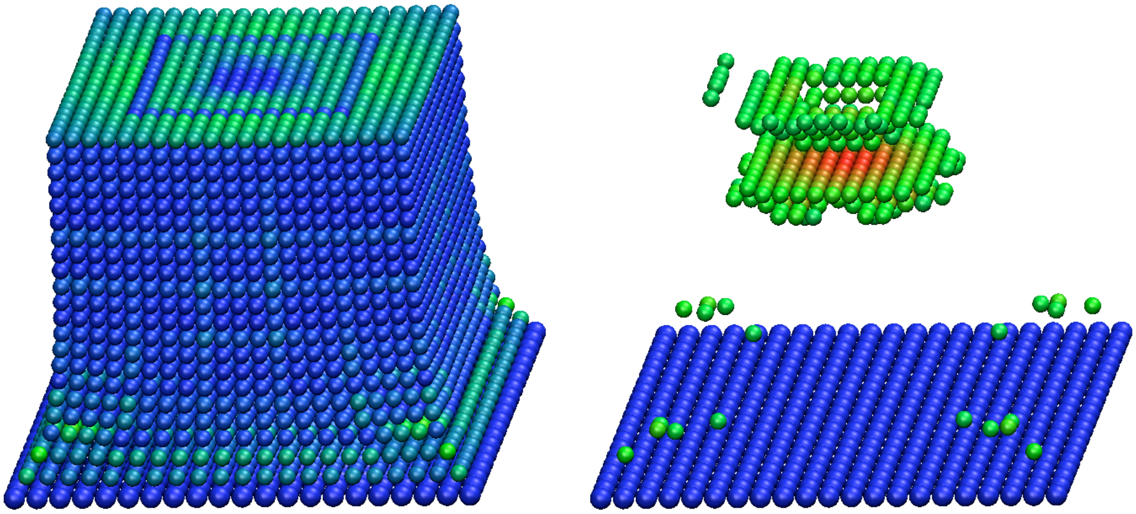


Figure 5.4: **(Left)** the residual $\mathcal{R}(\Pi\mathbf{u}_0, \hat{\mathbf{p}})$ corresponding to an Arlequin approximation of a $21 \times 21 \times 21$ uniform lattice. The position of the particles is in the equilibrated current configuration. The value of the normalized residual $\mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}})/\max_i \mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}})$, where $\mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}}) = (\partial E_i(\Pi\mathbf{u}_0)/\partial\mathbf{u}) \cdot \hat{\mathbf{p}}_i$, for each particle i is assigned a color in the figure, with red the highest values and blue the lowest. **(Right)** Here, the substrate particles are shown along with particles that have normalized residual $\mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}})/\max_i \mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}}) > 0.4$. This represents the region that most strongly contributes to the error in the quantity of interest. In this case, the error is primarily at the interface of the particle and continuum models, as expected.

Table 5.1: Results of an Arlequin approximation of a 21 × 21 × 21 uniform lattice with the quantity of interest being the average z-displacement of a 3×3 patch on the positive z-surface of the body.

$Q(\mathbf{u})$	-24.5
$Q(\Pi\mathbf{u}_0)$	-22.2
$\mathcal{R}(\Pi\mathbf{u}_0, \mathbf{p})$	-2.49
$\mathcal{R}(\Pi\mathbf{u}_0, \hat{\mathbf{p}})$	-2.17
$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\mathbf{p})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	1.08
$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\hat{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	0.94

Table 5.2: Results of an Arlequin approximation of a 21 × 21 × 21 polymer lattice with the quantity of interest being the average z-displacement of a 3×3 patch on the positive z-surface of the body.

$Q(\mathbf{u})$	-55.63
$Q(\Pi\mathbf{u}_0)$	-35.0
$\mathcal{R}(\Pi\mathbf{u}_0, \mathbf{p})$	-30.3
$\mathcal{R}(\Pi\mathbf{u}_0, \hat{\mathbf{p}})$	-14.8
$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\mathbf{p})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	1.47
$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\hat{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	0.72

Polymer Lattice, Arlequin Surrogate In this case, a 21 × 21 × 21 polymer lattice is taken and approximated by an Arlequin surrogate model. The Arlequin surrogate model consists of 174 trilinear elements, 17 trilinear Lagrange multiplier elements (for the overlap region), and 1183 particles. The constitutive equations for the continuum is the compressible Mooney-Rivlin material with coefficients given in Table 3.3.2. The shrinkage parameter $\lambda = 1.3$. The quantity of interest is the "slump": the average z-displacement of a 3 × 3 patch on the positive z-face of the body as in (5.8). Thus, this example serves to illustrate the effects of inhomogeneity in the lattice. Figure 5.5 displays the (projected) adjoint solution and Figure 5.6 the residual. Table 5.2 summarizes the results from the calculation. The effect of the inhomogeneity has effected the quality of the error estimate by approximately 20%. This is likely due to the fact that the projection is smooth in the overlap region where the actual particle positions in the base model are much more irregular. It

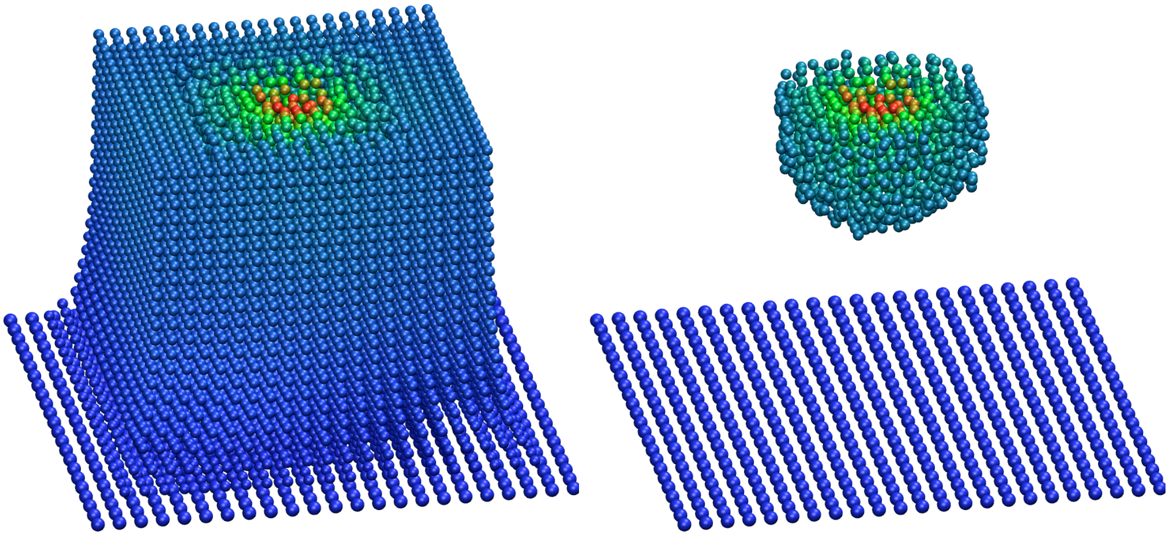


Figure 5.5: **(Left)** the adjoint solution $\hat{p}$ corresponding to an Arlequin approximation of a $21 \times 21 \times 21$ polymer lattice. The position of the particles is in the equilibrated current configuration. The value $\|\hat{\mathbf{p}}_i\| / \max_i \hat{\mathbf{p}}_i$ for each particle i is assigned a color in the figure, with red the highest values and blue the lowest. **(Right)** Here, the substrate particles are shown along with particles that have $\|\hat{\mathbf{p}}_i\| / \max_i \hat{\mathbf{p}}_i > 0.2$. This represents the region that most strongly influences the value of the quantity of interest.

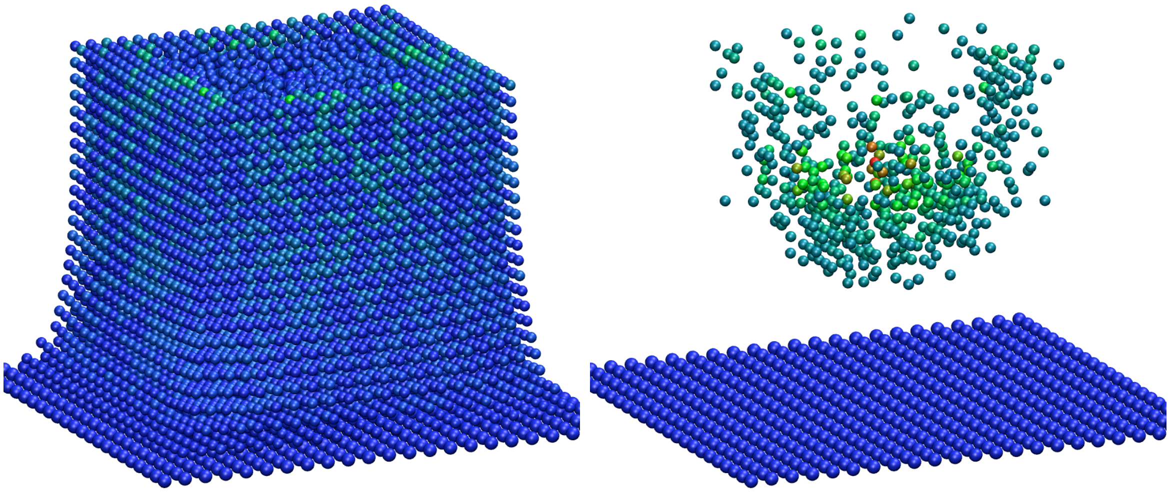


Figure 5.6: **(Left)** the residual $\mathcal{R}(\Pi\mathbf{u}_0, \hat{\mathbf{p}})$ corresponding to an Arlequin approximation of a 21 × 21 × 21 polymer lattice. The position of the particles is in the equilibrated current configuration. The value of the normalized residual $\mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}})/\max_i \mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}})$, where $\mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}}) = (\partial E_i(\Pi\mathbf{u}_0)/\partial\mathbf{u}) \cdot \hat{\mathbf{p}}_i$, for each particle i is assigned a color in the figure, with red the highest values and blue the lowest. **(Right)** Here, the substrate particles are shown along with particles that have normalized residual $\mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}})/\max_i \mathcal{R}_i(\Pi\mathbf{u}_0, \hat{\mathbf{p}}) > 0.25$. This represents the region that most strongly contributes to the error in the quantity of interest. In this case, the error is primarily at the interface of the particle and continuum models, as expected.

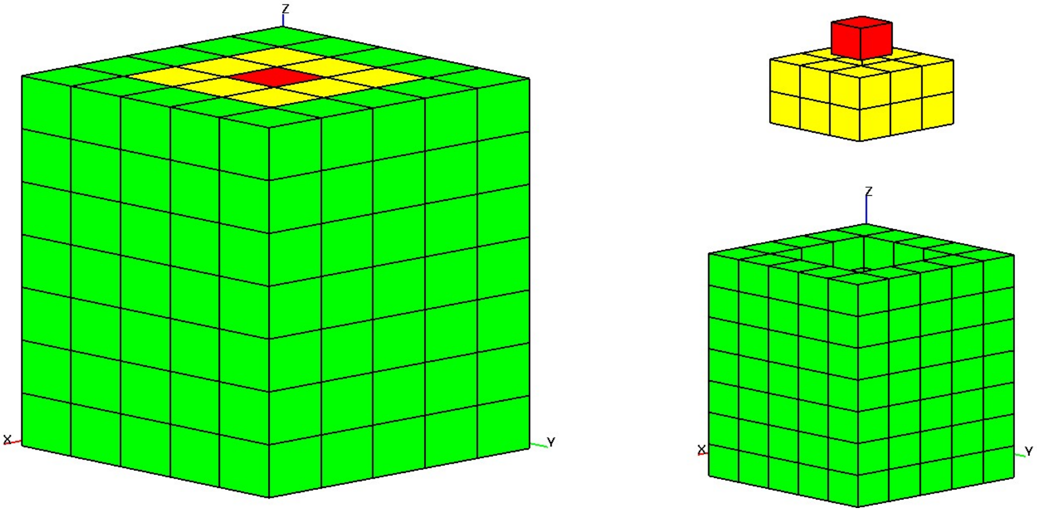


Figure 5.7: The partition of the Arlequin domain. Red cells denote regions modeled using the particle model, green are cell modeled using the continuum model, and yellow is the overlap between the two.

would seem that, for the polymer model, a more robust projection method could improve the error estimate, but this is not studied in the present work.

5.4 Adaptive Algorithm

The adaptive algorithm is now discussed in detail. First, the domain of the problem must be partitioned into cells that will be labeled by the model used within that region. For simplicity, the partition will coincide exactly with the finite element discretization of the continuum that covers the particle model. Figure 5.7 is an example of such a partition. Furthermore, the residual error estimate (5.5) must be partitioned in the same manner:

$$\mathcal{R}\left(\Pi \mathbf{u}_0, \hat{\mathbf{p}}\right) = \sum_{c=1}^{N_p} \mathcal{R}_c\left(\Pi \mathbf{u}_0, \hat{\mathbf{p}}\right) \tag{5.9}$$

where N_p is the number of partitioned cells, c labels each cell, and $\mathcal{R}_c\left(\Pi \mathbf{u}_0, \hat{\mathbf{p}}\right)$ is the value of the residual in the cell c. The value of $\mathcal{R}_c\left(\Pi \mathbf{u}_0, \hat{\mathbf{p}}\right)$ is computed by

$$\mathcal{R}_c\left(\Pi \mathbf{u}_0, \hat{\mathbf{p}}\right) = \sum_{i \in c} \sum_{j=1}^{n_i} \frac{\partial E_{ij}(\Pi \mathbf{u}_0)}{\partial \mathbf{u}} \cdot \hat{\mathbf{p}}_i \tag{5.10}$$

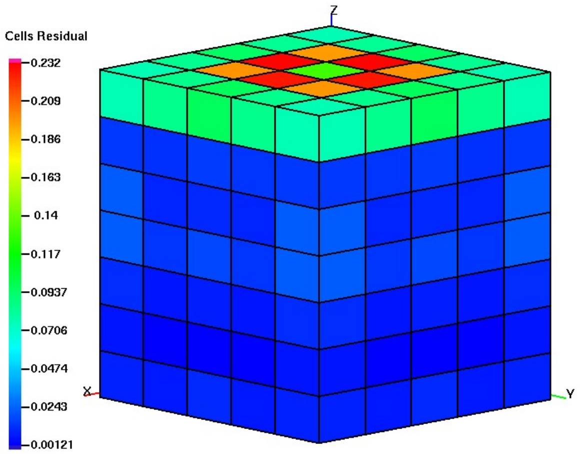


Figure 5.8: The residual partitioned over the domain.

Note that if the particle i is on a boundary between cells, it is scaled by the number of cells it overlaps. For instance, if a particle is on a face and belongs to two cells it is scaled by 1/2. Figure 5.8 shows an example of the distribution of error.

Thus, the adaptive algorithm proceeds as follows.

1. Specify the error tolerance for the quantity of interest, γ_{tol} and the refinement parameter α_a.

2. Solve the primal surrogate problem (1.3).

3. Compute the projection $\Pi\mathbf{u}_0$ of surrogate problem. Solve the adjoint problem (5.3).

4. Compute the residual (5.5). If $\frac{\mathcal{R}(\Pi\mathbf{u}_0,\hat{\mathbf{p}})}{Q(\mathbf{u})-Q(\mathbf{u}_0)} < \gamma_{tol}$, then stop. Otherwise, goto 5.

5. Partition the residual: compute $|\mathcal{R}_c(\Pi\mathbf{u}_0,\hat{\mathbf{p}})|$ for each cell c.

6. Relabel cells as "particle" cells if: $|\mathcal{R}_c(\Pi\mathbf{u}_0,\hat{\mathbf{p}})| > \alpha_a \max_c |\mathcal{R}_c(\Pi\mathbf{u}_0,\hat{\mathbf{p}})|$. That is, if a cell c has a residual a fraction α_a of the maximum, then that cell is "refined" and relabeled as particle.

7. Reconfigure overlap region to be consistent with new particle configuration.

8. Goto 2.

Table 5.3: Results of the Goals algorithm for a 21 × 21 × 21 uniform lattice with an Arlequin surrogate model. A total of five adaptive steps were needed to bring the error in the quantity of interest to within five percent.

Adaptive Step	$\frac{\mathcal{R}(\Pi \mathbf{u}_0,\hat{\mathbf{p}})}{Q(\Pi \mathbf{u}_0)}$	$\frac{\lvert\mathcal{R}(\Pi \mathbf{u}_0,\mathbf{p})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi \mathbf{u}_0)\rvert}$	$\frac{\lvert\mathcal{R}(\Pi \mathbf{u}_0,\hat{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi \mathbf{u}_0)\rvert}$
0	9.77%	108%	94.2%
1	8.24%	109%	93.8%
2	7.71%	110%	93.3%
3	6.75%	111%	91.9%
4	5.08%	116%	88.0%
5	4.74%	110%	92.1%

5.5 Numerical Examples: Adaptivity

Uniform Lattice, Arlequin Surrogate The same data are used here as was used for the example on error estimation. The adaptivity parameters $\alpha_a = 0.4$ and $\gamma_{tol} = 0.05$ were used. Table 5.4 lists the error and effectivity indices computed during the five adaptive steps. Figure 5.9 and Figure 5.10 show the configurations chosen by the adaptive algorithm and Figure 5.11 shows the partition of the residual for each configuration.

Polymer Lattice, Arlequin Surrogate The same data are used here as was used for the example in error estimation. The adaptivity parameters $\alpha_a = 0.4$ and $\gamma_{tol} = 0.05$ were used. Table 5.4 lists the error and effectivity indices computed during the five adaptive steps.

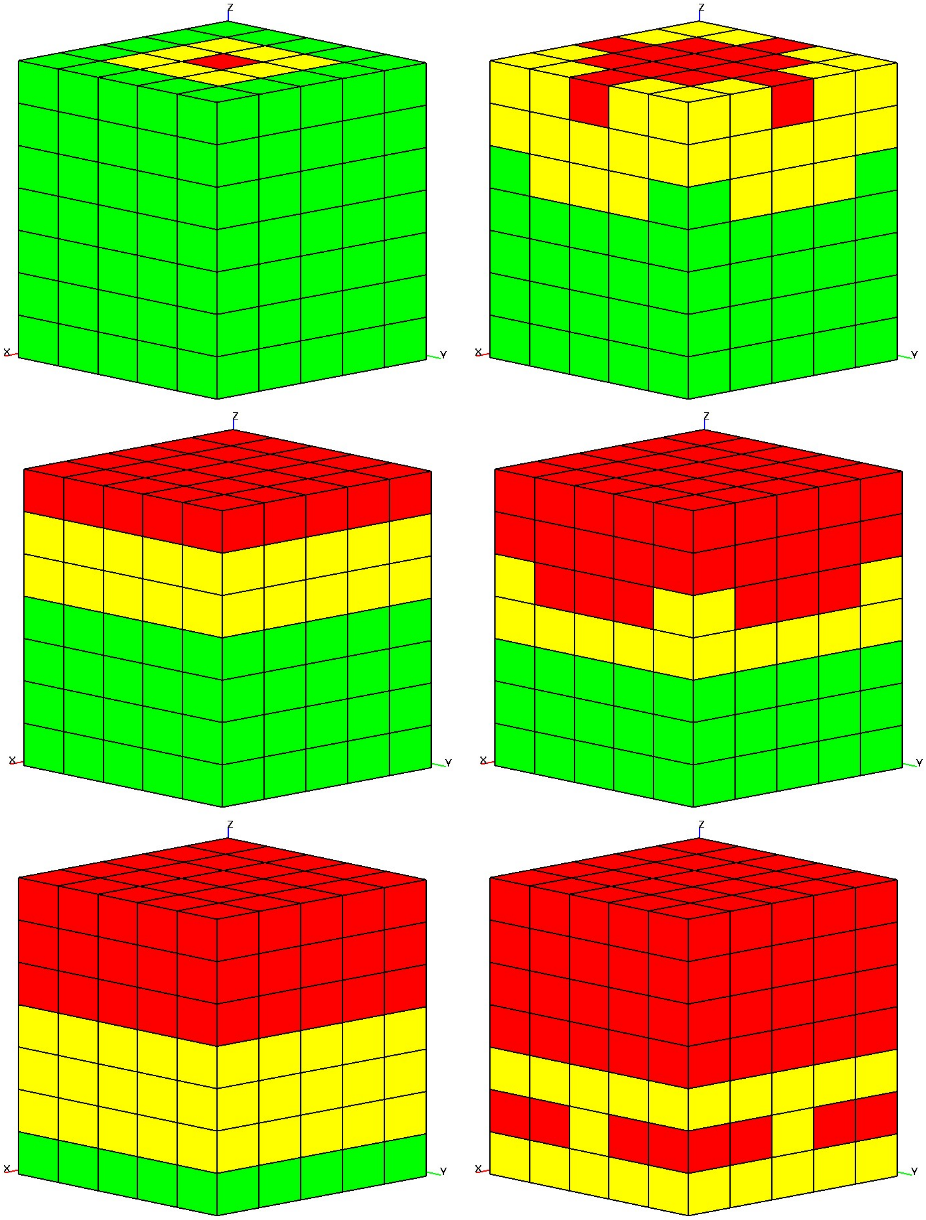


Figure 5.9: The configurations chosen by the adaptive algorithm for a 21 × 21 × 21 uniform lattice with an Arlequin surrogate model and $\alpha_a = 0.4$ and $\gamma_{tol} = 0.05$. Red cells denote particle model regions, green cells the continuum model, and yellow cells denote the overlap regions.

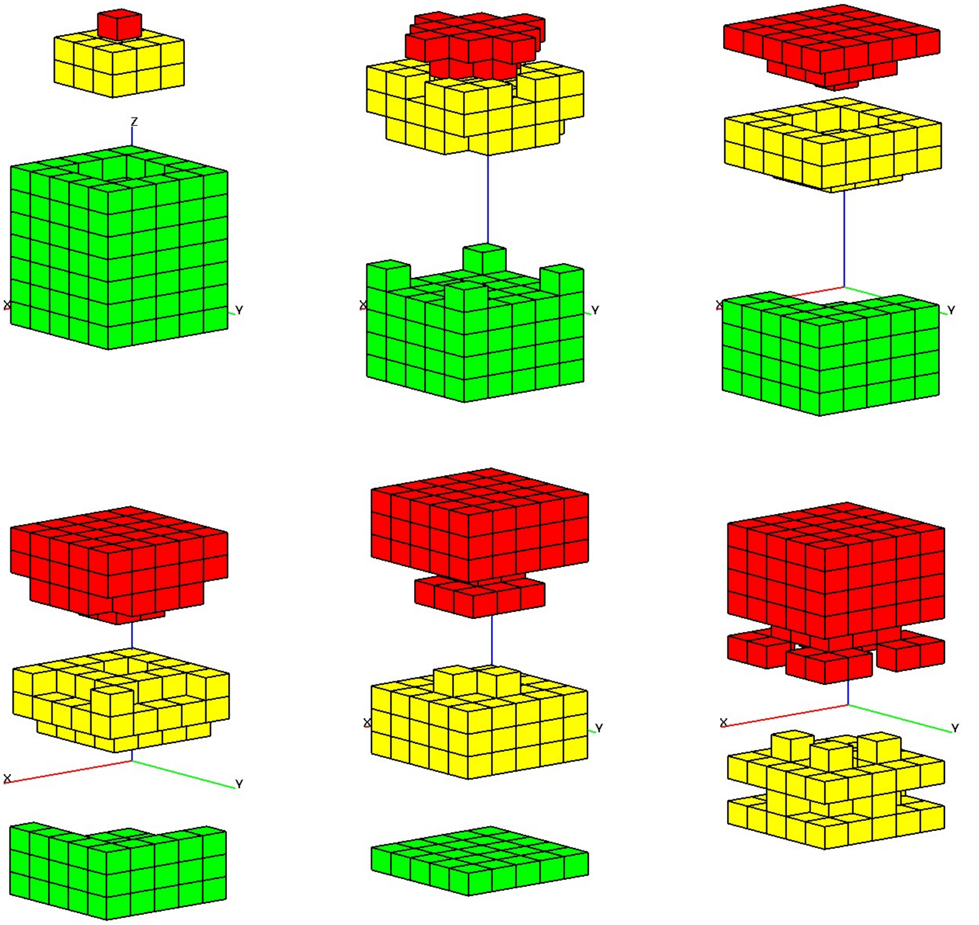


Figure 5.10: Exploded view of the configurations chosen by the adaptive algorithm for a 21 × 21 × 21 uniform lattice with an Arlequin surrogate model and $\alpha_a = 0.4$ and $\gamma_{tol} = 0.05$. Red cells denote particle model regions, green cells the continuum model, and yellow cells denote the overlap regions.

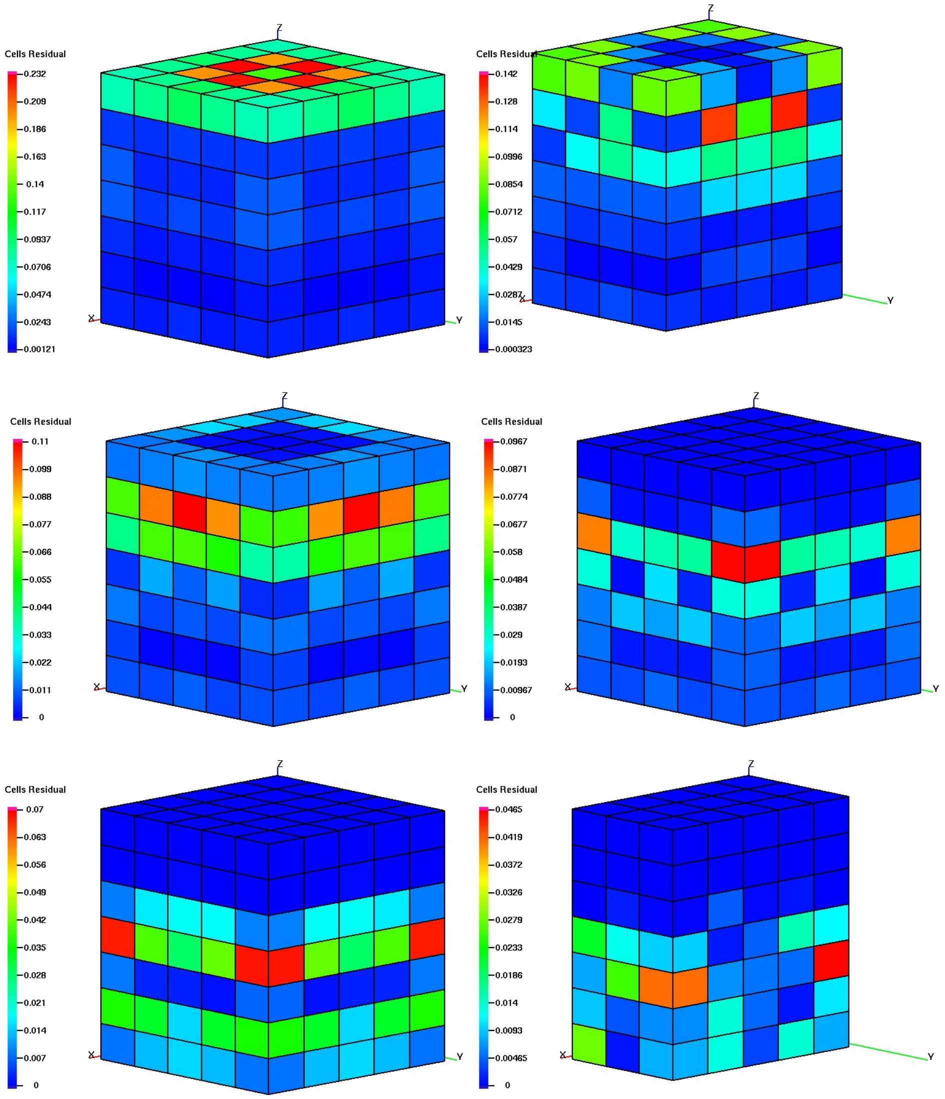


Figure 5.11: The residual partition for each step of the adaptive algorithm for a $21 \times 21 \times 21$ uniform lattice with an Arlequin surrogate model and $\alpha_a = 0.4$ and $\gamma_{tol} = 0.05$. In some cases, the image was sliced to expose regions containing the greatest amount of error.

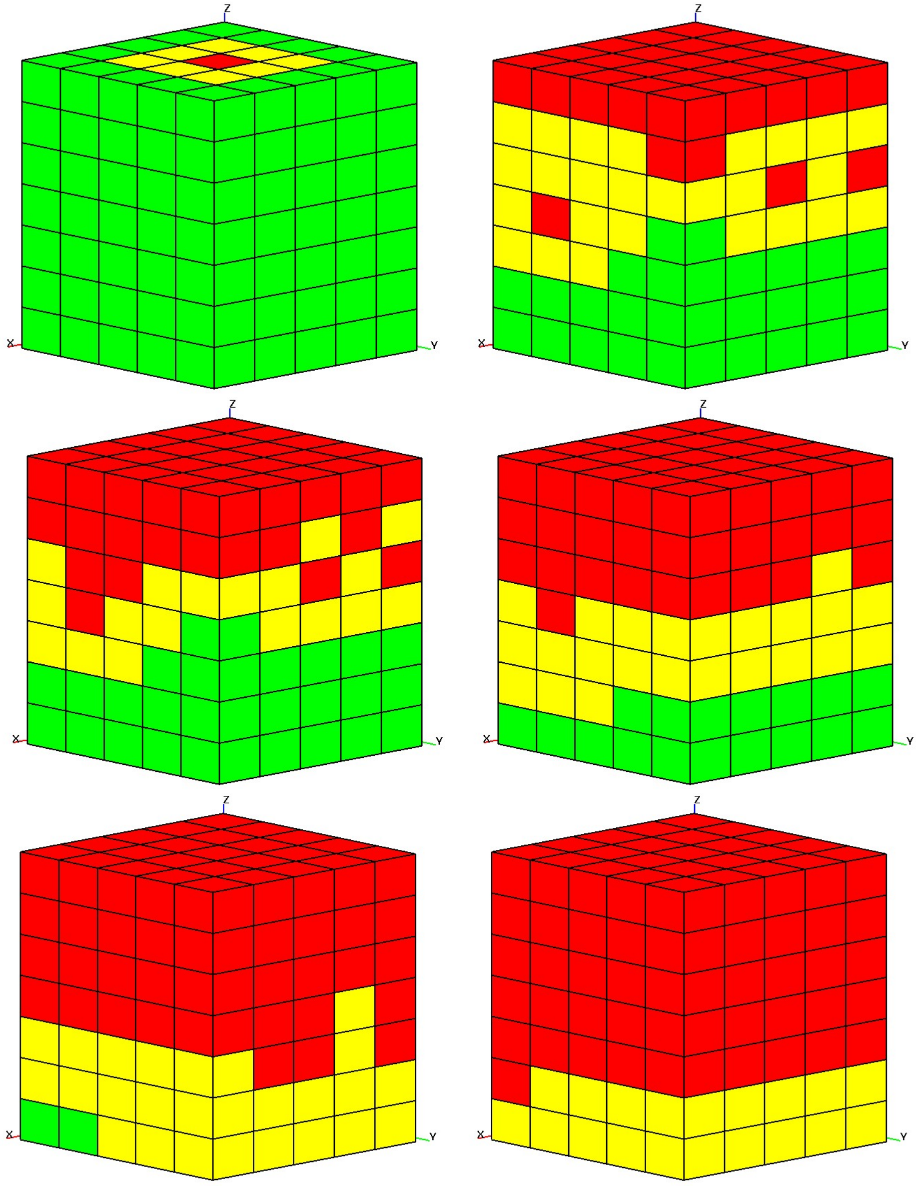


Figure 5.12: The configurations chosen by the adaptive algorithm for a 21 × 21 × 21 polymer lattice with an Arlequin surrogate model and $\alpha_a = 0.4$ and $\gamma_{tol} = 0.06$. Red cells denote particle model regions, green cells the continuum model, and yellow cells denote the overlap regions.

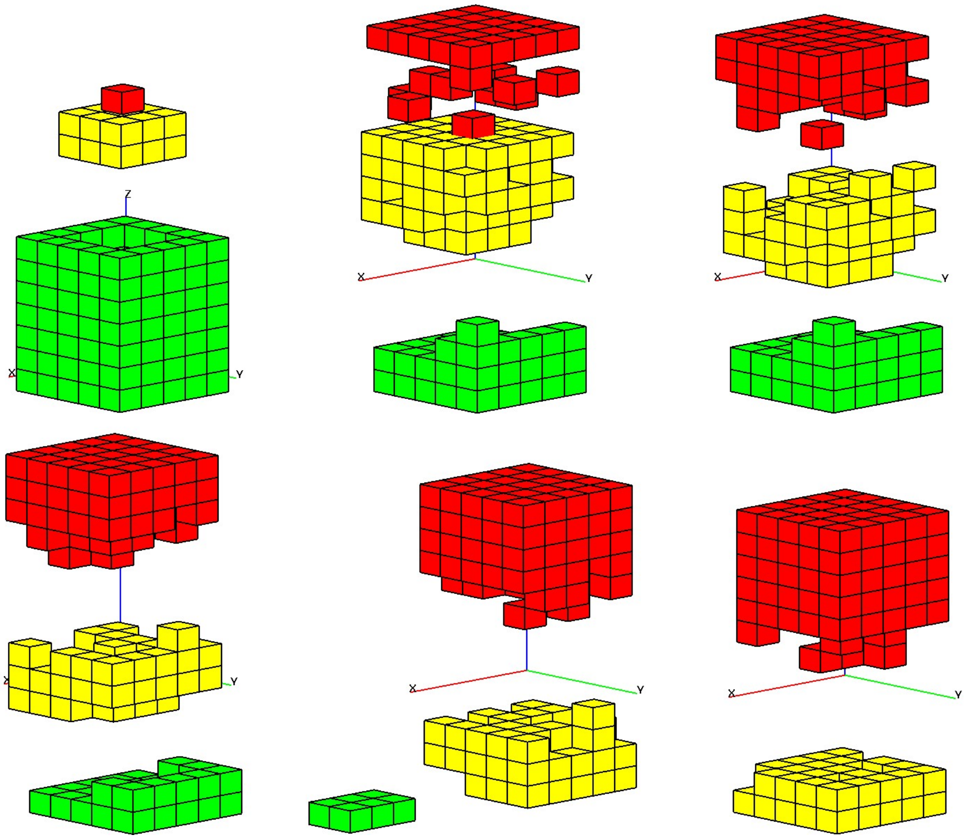


Figure 5.13: Exploded view of the configurations chosen by the adaptive algorithm for a $21 \times 21 \times 21$ polymer lattice with an Arlequin surrogate model and $\alpha_a = 0.4$ and $\gamma_{tol} = 0.06$. Red cells denote particle model regions, green cells the continuum model, and yellow cells denote the overlap regions.

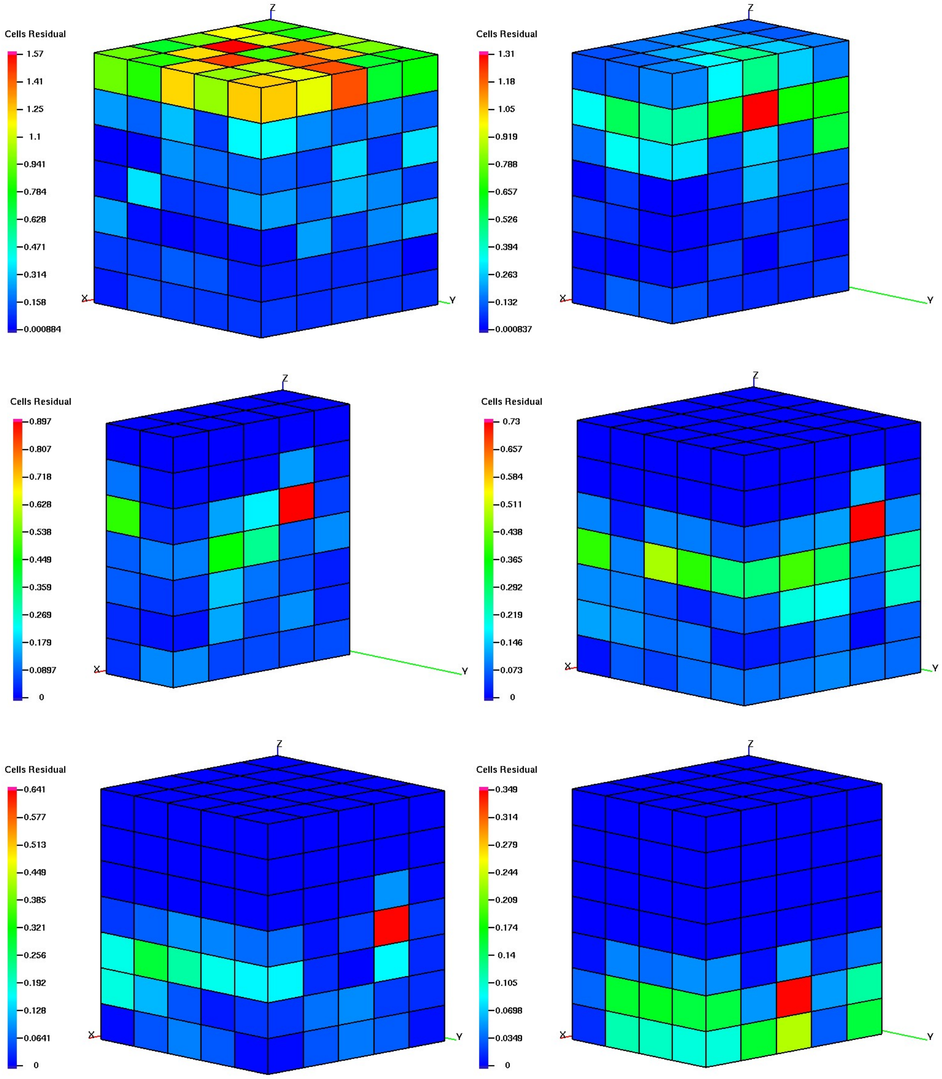


Figure 5.14: The residual partition for each step of the adaptive algorithm for a $22 \times 22 \times 22$ polymer lattice with an Arlequin surrogate model and $\alpha_a = 0.4$ and $\gamma_{tol} = 0.05$. In some cases, the image was sliced to expose regions containing the greatest amount of error.

Table 5.4: Results of the Goals algorithm for a $21 \times 21 \times 21$ polymer lattice with an Arlequin surrogate model. A total of five adaptive steps were needed to bring the error in the quantity of interest to within six percent.

Adaptive Step	$\frac{\mathcal{R}(\Pi\mathbf{u}_0,\hat{\mathbf{p}})}{Q(\Pi\mathbf{u}_0)}$	$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\mathbf{p})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\hat{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$
0	42.2%	147%	71.6%
1	30.8%	141%	67.4%
2	18.6%	152%	59.2%
3	13.5%	156%	51.7%
4	10.2%	145%	48.7%
5	5.34%	137%	36.8%

Chapter 6

Approximation of Adjoint Solution and Residual Calculation

6.1 Introduction

In the previous chapter, strategies to project the Arlequin solution to the full lattice space, solve the adjoint problem, and compute the residual are discussed in detail. Furthermore, an adaptive algorithm is introduced. However, in practical applications, alternative techniques must be considered to compute both the adjoint and residual for error estimation as the systems of interest will be too large to use the full lattice. Such strategies are discussed in the current chapter followed by numerical results illustrating the accuracy of the error estimates and their effectiveness within the Goals adaptive modeling algorithm.

6.2 Adjoint and Residual Approximations

Recall from the previous chapter that in order to compute error estimates in quantities of interest using the Arlequin surrogate solutions, a projection operator Π was introduced that mapped the mixed particle-continuum model to the full particle space. However, for applications of engineering interest, the cost of computing the adjoint solution evaluating the residual would be prohibitive. Furthermore, it would be ineffective to compute the dual and residual with the same surrogate model since:

$$B_0(\mathbf{u}_0; \mathbf{v}_0) = F_0(\mathbf{v}_0) \quad \forall \mathbf{v}_0 \in V_0 \tag{6.1}$$

so that

$$\mathcal{R}_0(\mathbf{u}_0; \mathbf{p}_0) = F_0(\mathbf{p}_0) - B_0(\mathbf{u}_0; \mathbf{p}_0) = 0 \tag{6.2}$$

Thus, the calculation of the adjoint and residual must involve more information than is present in the surrogate model, but still be cost efficient to be a practical tool.

The strategy proposed here is to enrich the surrogate model by enlarging the region of the base model *only for the calculation of the adjoint and residual.* This is an analogous strategy used in [58] and was briefly summarized in Section 1.5.1. In the case of the Arlequin method, this simply means relabeling neighboring cells as particle regions. The parameter in this algorithm is the number of layers to enrich the model. A one-layer enrichment corresponds to relabeling all cells as particle regions that directly neighbor the current particle regions. Two layers would transform the nearest neighbors *and* the next nearest neighbors. Figure 6.2 illustrates the idea of enrichment.

Once cells have been reconfigured, then a projection must be applied so that the surrogate solution is in the correct space. This projection is labeled $\tilde{\Pi} : (X, M) \rightarrow (\tilde{X}, \tilde{M})$ and is discussed subsequently. First, the surrogate solution corresponding to continuum or particle cells that were unchanged by the enrichment procedure are left unchanged; in other words, the value of the displacements are copied to the new vector. Continuum cells that were converted to particle or overlapping cells require the addition of particles: this is accomplished using the procedure discussed in Section 5.2. The caveat here is that, unlike the case of the projection Π, the Lagrange multiplier cannot be ignored. Indeed, Figure 6.2 shows that this procedure will *always* change the overlap region where the Lagrange multiplier is defined.

To understand the consequences of choosing this projection on the calculation of the error estimate, the one-dimensional problem from Section 4.2 is examined. In this case,

$$\begin{aligned} \tilde{V}_c &= \left\{ v \in H^1(\tilde{\Omega}_c) : v(0) = 0 \right\} \\ \tilde{V}_d &= \left\{ z \in \mathbb{R}^{\tilde{m}+1} \right\} \\ \tilde{X} &= \tilde{V}_c \times \tilde{V}_d \end{aligned} \tag{6.3}$$

and

$$\tilde{M} = \begin{cases} H^1(\tilde{\Omega}_o)/\mathbb{R}, & \text{if } \beta_1 = 0 \\ H^1(\tilde{\Omega}_o), & \text{otherwise} \end{cases} \tag{6.4}$$

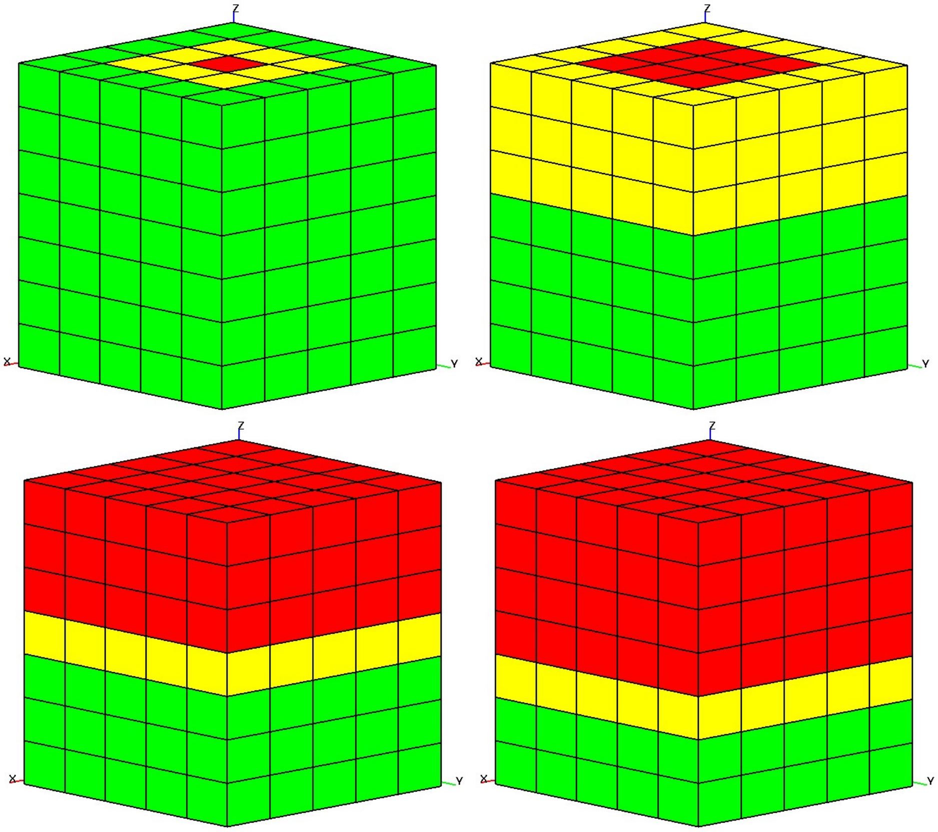


Figure 6.1: Illustration of the enrichment procedure used to approximate the calculation of the adjoint solution and the evaluation of the residual. A sequence of one, two, and three layer enrichments are shown here. Red cells denote particle model regions, green cells the continuum model, and yellow cells denote the overlap regions.

where the $\tilde{(\cdot)}$ symbols correspond with the enriched configuration. Then, the (enriched) adjoint problem is

$$\begin{aligned} &\text{Find } \tilde{P} \in \tilde{X},\ \tilde{p}_\lambda \in \tilde{M} \text{ such that:} \\ \tilde{a}(\tilde{V}, \tilde{P}) + b(\tilde{p}_\lambda, \tilde{V}) &= Q(\tilde{V}) \quad \forall \tilde{V} \in \tilde{X} \\ b(\tilde{\mu}, \tilde{P}) &= 0 \qquad\ \ \forall \tilde{\mu} \in \tilde{M} \end{aligned} \tag{6.5}$$

where $\tilde{P} = (\tilde{p}_c, \tilde{p}_d)$ and $\tilde{p}_c, \tilde{p}_d, \tilde{p}_\lambda$ are the continuum, discrete, and Lagrange multiplier components of the adjoint solution. Thus, it is clear that the adjoint solution satisfies the constraint on the overlap region. In particular, letting $\tilde{\Pi}\lambda$ correspond to the (at this point, arbitrary) choice of the projection of the Lagrange multiplier,

$$\begin{aligned} \tilde{\mathcal{R}}(\tilde{\Pi}U, \tilde{P}) &= \tilde{a}(\tilde{\Pi}U, \tilde{P}) + b(\tilde{\Pi}\lambda, \tilde{P}) + b(\tilde{p}_\lambda, \tilde{U}) \\ &= \tilde{a}(\tilde{\Pi}U, \tilde{P}) + b(\tilde{p}_\lambda, \tilde{\Pi}U) \end{aligned} \tag{6.6}$$

since $b(\tilde{\mu}, \tilde{P}) = 0\ \forall \tilde{\mu} \in \tilde{M}$. Thus, the choice of $\tilde{\Pi}\lambda$ has no effect on the calculation of the error estimate for the quantity of interest.

Although the constraint is strongly enforced in the continuous case, the constraint is only weakly enforced in the discrete case. Therefore, the constraint will not necessarily be enforced element-wise. The implication here is on the adaptive algorithm. When the residual is partitioned over the domain, although the effects of the term $b(\tilde{\lambda}, \tilde{P})$ will cancel when the residual partition is summed, it will locally pollute the residual value. For the numerical results presented, the local pollution is on the order of 1%-5% of the value of the cell residual $\mathcal{R}_c(\tilde{\Pi}\mathbf{u}_0, \tilde{\mathbf{p}})$. Therefore, this effect is small and will be neglected.

6.3 Numerical Results

The $21 \times 21 \times 21$ uniform lattice considered previously is again studied here. Table 6.1 shows the effectivity indices for various levels of enrichment for an error estimate where the adjoint is computed on the enriched configuration, but then is projected to the full lattice space so that there is no approximation in the residual calculation. This is to be compared with the estimator where both dual and residual are computed on the enriched configuration. This represents the computable error estimate that would

Table 6.1: Effectivity indices for various levels of enrichment using the approximate adjoint, and then both approximate residual and adjoint on a 22 × 22 × 22 uniform lattice with an Arlequin surrogate. The indices computed using the exact dual and residual are inserted for comparison.

$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\mathbf{p})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\hat{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	Number of Enrichments	$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\Pi\tilde{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	$\frac{\lvert\tilde{\mathcal{R}}(\tilde{\Pi}\mathbf{u}_0,\tilde{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\tilde{\Pi}\mathbf{u}_0)\rvert}$
1.08	0.94	5	0.69	0.74
		4	0.58	0.57
		3	0.53	0.37
		2	0.45	0.26
		1	0.36	0.15

Table 6.2: Effectivity indices for various levels of enrichment using the approximate adjoint, and then both approximate residual and adjoint on a 22 × 22 × 22 polymer lattice with an Arlequin surrogate. The indices computed using the exact dual and residual are inserted for comparison.

$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\mathbf{p})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\hat{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	Number of Enrichments	$\frac{\lvert\mathcal{R}(\Pi\mathbf{u}_0,\Pi\tilde{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\Pi\mathbf{u}_0)\rvert}$	$\frac{\lvert\tilde{\mathcal{R}}(\tilde{\Pi}\mathbf{u}_0,\tilde{\mathbf{p}})\rvert}{\lvert Q(\mathbf{u})-Q(\tilde{\Pi}\mathbf{u}_0)\rvert}$
1.47	0.72	5	0.64	0.67
		4	0.59	0.57
		3	0.48	0.43
		2	0.39	0.32
		1	0.23	0.16

be utilized in practice. Analogous results for the 21 × 21 × 21 polymer lattice are shown in Table 6.2. Note that in both cases, enrichments greater than five levels will lead to dual and residual calculations on the full lattice space.

As the results show, the fidelity of the error estimate can be severely compromised depending on the enrichment level used for the error estimate. This is particularly true in the uniform case. In the polymer case, the even moderate level of enrichments yield error estimates within 15% of the error estimates without approximating the dual or residual. This may be due to the fact that the quantity of interest is more global in nature and that quantities more local will yield better results. This requires further investigation.

Chapter 7

Summary and Directions for Future Research

The development of computational tools to model physical phenomena that occur across many spatial scales is an essential step in aiding the design and implementation of the manufacturing of next generation of computer hardware. Such tools can provide analysts insight to problems which are otherwise infeasible to resolve through experiments or due to cost or physical limitations. This work represents a step in the development of such tools.

The objective of this dissertation is to construct an adaptive multiscale model of molecular particle models whose solution is rooted in goal-oriented error estimation and adaptive modeling techniques. This entailed the specification of a base model of molecular statics of polymers, the coarse graining of the particle model to a nonlinear elastic continuum model, the coupling of the two via the Arlequin method, estimating the error incurred by using such surrogates, and adapting the model to control the error in specific quantities of interest.

The base model is based upon a lattice model of polymers where the polymer network is constructed using a monte-carlo type algorithm that models the chemical kinetics of the material. Harmonic potentials are assigned to bonds that are formed during this process from which a molecular statics problem is posed. It is observed that not only is the immense size of realistic simulations a limiting factor of the model, but also is the challenge of efficiently finding a solution. Indeed, as the problem size grows, so also does the number of Newton iterations.

Based on arguments from statistical mechanics, as well as experimental evidence, a nonlinear elastic continuum model is chosen as a coarse grain representation of the base model. Numerical experiments on representative volume elements (RVE's) are used to determine coefficients for the continuum model so that it is as compatible as possible with the underlying particle model. Furthermore, a simple augmentation to the classical finite element approximation is introduced to account for the inherent initial strain present in the molecular model.

With the particle and continuum models properly defined, the next step is to couple the two to construct the surrogate model used to approximate the base model. The coupling here is based on the Arlequin method where Lagrange multipliers enforce constraints on the displacements and derivatives on a region of overlap between the particle and continuum model. A one-dimensional model problem is constructed where a system of harmonic springs are coupled to a linearly elastic rod. It is shown that this Arlequin problem is well-posed. Several numerical experiments illustrate the mathematical findings. This surrogate is also implemented in a three-dimensional setting where the base polymer model is coupled to the nonlinear elastic continuum model developed previously. It is shown that the surrogate model provides a substantial cost benefit in both computer resources required as well as simulation time. This is due not only to smaller systems being solved, but also to the effect of the continuum model smoothing energy landscape of the particle model.

Finally, the Goals framework is implemented on top of the Arlequin surrogate that is constructed. Error estimates are computed for several numerical examples where the dual and residual are computed exactly, as well as the case where the surrogate primal solution is used in the solution of adjoint problem. For uniform lattices, the results are very good with effectivity indices on the order of $0.92-1.08$. While the results are still acceptable for the polymer case, the effectivity indices are in the range $0.72 - 1.4$. The error estimates are then used to drive an adaptive algorithm where the residual is partitioned over the domain and the cells are selected for refinement, i.e. the model becomes the particle model in the cells where the contributions to the error are the largest. It is seen that the adaptive algorithm controlled the error to within the specified tolerance although the refinement procedure had to refine a great deal. This indicates that either the quantity of interest is global in nature or the models are incompatible.

The same procedure is reiterated, but now with error estimates computed with approximate adjoint solutions and residual calculations. This idea is based on enriching the current model partition to include more of the base model just for the calculation of the dual and residual so as to have enough information to obtain a reasonable error estimate, but at much less cost that obtaining the full adjoint and residual. As expected, for low enrichment, the effectivity indices suffer, but as the model is enriched further, the estimates of the error became comparable to those obtained with the exact dual and residual calculations. This is especially true in the case of the polymer model.

Although major strides have been made towards developing practical tools for adaptive multiscale modeling of polymers, there is still much to be done. Among issues and questions that need to be addressed in future work are the following:

1. The base model deserves a more careful development. What are realistic potentials for the molecular model? What potentials would allow the breaking of bonds to simulate template separation?

2. The incorporation of stochasticity into the model is an important extension. In particular, how does the inherent randomness of the bonds affect the fidelity of the error estimates? How does one incorporate randomness into the coupled model? What is the uncertainty in the continuum model due to the randomness of the molecular model? How does uncertainty in the potentials affect the outcome of the model?

3. The development of more robust solution methods for this class of problems is needed. As was seen, while the coupled model performs well for uniform lattices or polymer lattices of smaller dimension, the convergence can become effected once the particle region becomes large. This is due to the roughness of the energy landscape for high-dimensional particle models and the inability of a nonlinear equations algorithm to cope with it. This may also include more effective pre-conditioners for the iterative linear solvers.

4. The construction of alternative coupling formulations deserves further study. Although stability of the solution is mesh independent, the accuracy certainly is not. In particular, the Lagrange multiplier element size strongly affects the character of the solution as the mesh size became comparable to the particle spacing. Novel formulations perhaps based on RVE size may provide more elegant solutions.

5. A deeper investigation into the error estimates can be made. In particular, why the error estimates for the polymer model are inferior to those of uniform lattices is an open problem. Their resolution may involve developing novel techniques to be implemented during the enrichment phase of the adjoint and residual calculations.

6. Developing further the computational infrastructure to handle more general scenarios is an important area for future research. This work has been restricted to cubic lattices and meshes and very specific constraints on the overlap region coinciding with particles. A more functional tool would not have these restrictions. Furthermore, the parallel implementation of the coupling and adaptive algorithms for large scale calculations would provide the opportunity for true multiscale simulation.

Appendix A

Reactive Ion Etching Process

The large number of etching techniques commercially available can be represented in two broad classes: *wet etching* and *dry etching*. Wet etching refers to the use of liquid chemicals to remove material solely through chemical reactions. Dry etching is a process whereby molecules on the solid surface are ejected by gaseous materials through physical bombardment, by chemical reactions with reactive species at the surface, or a combination of these two mechanisms. The focus here will be on the dry etching process *Reactive Ion Etching*. The process is illustrated schematically in Fig. A.1.

The wafer(s) is set on a conducting plate in a chamber. Another conducting plate is suspended above the sample. A gas is pumped into the chamber between the conducting plates and is converted into a plasma[1]. An electric potential is produced across the two conducting plates so that the charged particles in the plasma are accelerated towards the wafer sample. If enough energy is supplied, the momentum of the charged particle dislodges material. This physical process is called *sputtering*. Sputtering is effective in *uniformly* removing material from the face of the sample while maintaining the aspect ratio of the feature (assuming enough energy is supplied to maintain nearly vertical momentum in the charged particles). For example, this process could be used for the breakthrough etch mentioned in the Section 1.3. However, the cost of supplying the necessary energy is often very high and the duration of the process can be very long. Therefore, the process is complemented by chemical reactions; this is accomplished by choosing a plasma gas that can form volatile by-

---

[1] A plasma is an ionized gas consisting of positive ions and free electrons so that there is almost no overall electrical charge

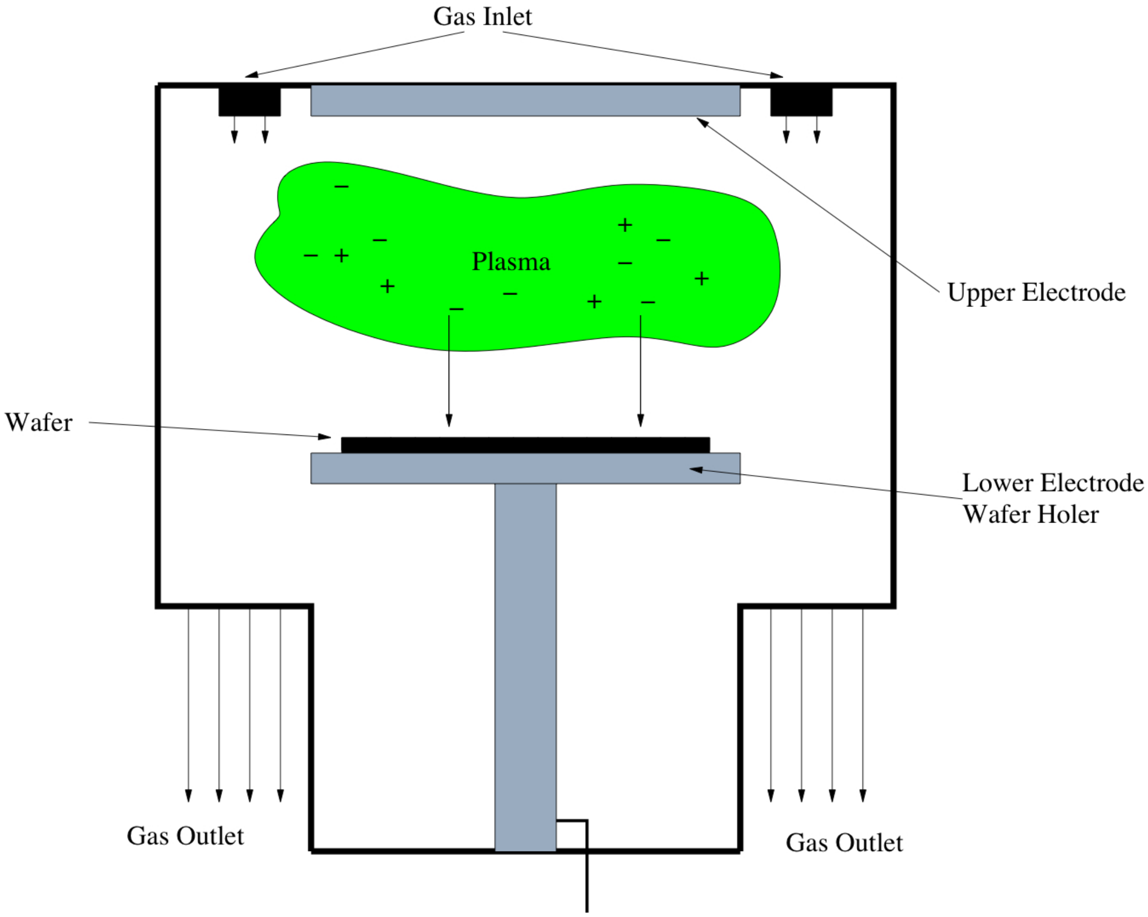


Figure A.1: Schematic of Reactive Ion Etching (RIE) chamber.

products. This produces *anisotropic* etch rates while, at the same time, decreasing the overall etch time. In other words, the accelerated radicals will chemically react with the etching material to form by-products that can be easily removed. And, if two different materials are present, the rate of reaction may be different for the two materials resulting in different etch rates for each.

The occurrence of different etch rates for distinct materials is particularly important for the transfer etch and the final substrate etch. For example, in the final etch, the Si substrate is etched simultaneously with the polymer etch barrier and transfer layer. The polymers restrict the etching of Si in a way to maintain the desired imprint pattern. Thus, it is essential that the etch rate for Si be much higher than that of the polymer. The ratio of such rates is typically 10:1 to 20:1.

The two most common choices of the plasma gas are O and CF_4, O being used for etching purely organic compounds[2] so that the by products are CO_2 and H_2O. These by-products are easily removed by a vacuum pump. However, if Si is present in the etching material, then SiO_2 may be formed. This can be difficult to remove and, in some cases, may refill etched areas. The by-products when using CF_4 are gaseous SiF_2 and SiF_4. Again, these by-products are easily removed by a vacuum pump.

---

[2] Organic compounds are molecules containing only C, O, and N atoms.

Appendix B

Review of Chemical Kinetics

The subject of chemical kinetics addresses the rates of chemical reactions and how the rates are affected by factors such as concentration and temperature. A primary branch of chemical kinetics is *macroscopic kinetics*, which relates the bulk behavior of very large groups of molecules in thermal equilibrium. These macroscopic investigations provide insight to molecular processes through experimental testing of hypotheses. *Microscopic kinetics* provides direct insight to molecular processes through controlled experimental procedures carried out on substances in particular states. The concepts discussed here fall under macroscopic kinetics.

The stoichiometry of a chemical reaction is represented as

$$a\mathrm{A} + b\mathrm{B} + \cdots \to \cdots + y\mathrm{Y} + z\mathrm{Z} \tag{B.1}$$

where capital letters are symbols for chemical substances (molecules or elements) and lower case letters denote the *stoichiometric* coefficient of each substance. For reactants, the stoichiometric coefficient is negative, while for products it is positive. For example, consider the following equation:

$$\mathrm{N_2} + 3\mathrm{H_2} \to 2\mathrm{NH_3} \tag{B.2}$$

The stoichiometric coefficient for NH_3 is 2 while for N_2 it is -1 and -3 for H_2.

When the stoichiometric equation encapsulates the entire reaction for the formation of the products from reactants, it is said to exhibit *time-independent* stoichiometry. That is, no intermediate reactions are required to describe the chemical transition. If intermediates are formed during the reaction, then the reaction is said to exhibit *time-dependent* stoichiometry. For example, the overall reaction for the thermal decomposition of acetone can be represented as

$$2\mathrm{CH_3COCH_3} \rightarrow 2\mathrm{CH_4} + \mathrm{C_2H_4} + 2\mathrm{CO} \tag{B.3}$$

However, it is observed that the reaction proceeds as

$$\begin{aligned} \mathrm{CH_3COCH_3} &\rightarrow \mathrm{CH_2CO} + \mathrm{CH_4} \\ 2\mathrm{CH_2CO} &\rightarrow \mathrm{C_2H_4} + 2\mathrm{CO} \end{aligned} \tag{B.4}$$

Thus, this reaction possesses time-dependent stoichiometry.

For reactions exhibiting time-independent stoichiometry, a convenient parameter is the *extent of reaction* ξ defined as

$$\xi \equiv \frac{n - n_0}{\nu} = \frac{\Delta n}{\nu} \tag{B.5}$$

where n is the amount of the substance at some time t, n_0 the amount at the initial time, and ν is the stoichiometric coefficient. Conservation of mass requires that, for any products Y and Z formed during a reaction, such as that in (B.1)

$$\xi = \frac{\Delta n_{\mathrm{Y}}}{y} = \frac{\Delta n_{\mathrm{Z}}}{z} \tag{B.6}$$

and similarly for reactants. That is, the amount of product formed must be the same as that of the reactant consumed in proportion to the stoichiometric coefficients. Note that this is not the case for time-dependent stoichiometry due to the presence of intermediates.

The extent of reaction provides the basis for the definition of the *rate of reaction*:

$$v \equiv \frac{1}{V}\frac{d\xi}{dt} = \frac{\dot{\xi}}{V} \tag{B.7}$$

where V is the volume occupied by all chemical species taking part in the reaction. From the definition of the extent of reaction, the rate of reaction of a substance n_i

with stoichiometric coefficient ν_i is then

$$v = \frac{1}{\nu_i V}\frac{dn_i}{dt} \tag{B.8}$$

Thus, for the general reaction (B.1),

$$v = -\frac{1}{aV}\frac{dn_A}{dt} = -\frac{1}{bV}\frac{dn_B}{dt} = \frac{1}{yV}\frac{dn_Y}{dt} = \frac{1}{zV}\frac{dn_Z}{dt} \tag{B.9}$$

If the volume V is constant, then n_i/V is the concentration of the substance, so that the rate of reaction v becomes

$$v = -\frac{1}{a}\frac{d[A]}{dt} = -\frac{1}{b}\frac{[B]}{dt} = \frac{1}{y}\frac{[Y]}{dt} = \frac{1}{z}\frac{d[Z]}{dt} \tag{B.10}$$

where the brackets [ ] denote concentrations of the various substances, often given in units of moles per liter.

For a chemical reaction, such as $aA + bB \rightarrow yY + zZ$, the rate of reaction is often assumed to be expressible empirically in the form

$$v = k[A]^\alpha[B]^\beta \tag{B.11}$$

where k, α, and β are independent of concentration and time. k is the experimentally determined *rate constant.* The exponents α and β are the *orders of reaction* for each of the respective concentrations. The *total order of reaction* is the sum of the exponents. These parameters are also empirically determined. For example, if the rate of reaction of the form

$$v = k[A] \tag{B.12}$$

then the reaction is said to be first order. A second order reaction has the form

$$v = k[A]^2 \tag{B.13}$$

or

$$v = k[A][B] \tag{B.14}$$

Appendix C

Derivation of Hessians for Molecular Potentials

C.1 Harmonic Potential

The harmonic potential is represented as

$$\begin{aligned} E_{ij}^{H} &= \frac{1}{2}k_{ij}\left(\|\mathbf{x}_j - \mathbf{x}_i\| - r_{ij}^{0}\right)^2 \\ &= \frac{1}{2}k_{ij}\left(r_{ij} - r_{ij}^{0}\right)^2 \end{aligned} \tag{C.1}$$

where k_{ij} and r_{ij}^0 are the constants for the spring connecting i and j, r_{ij} is the distance between particles i and j, and $\|\cdot\|$ is the standard Euclidean norm. Thus,

$$\frac{\partial E_{ij}^{H}}{\partial r_{ij}} = k_{ij}\left(r_{ij} - r_{ij}^{0}\right) \tag{C.2}$$

so that, using (2.23),

$$\mathbf{g}_{ij} = k_{ij}\left(\|\mathbf{x}_j - \mathbf{x}_i\| - {r_{ij}}^{0}\right)\frac{(\mathbf{x}_j - \mathbf{x}_i)}{\|\mathbf{x}_j - \mathbf{x}_i\|} \tag{C.3}$$

By definition,

$$\mathbf{H}_{ij} = \frac{\partial \mathbf{g}_{ij}}{\partial \mathbf{x}} \equiv \lim_{\theta \to 0}\frac{1}{\theta}\left[\mathbf{g}_{ij}\left(\mathbf{x} + \theta\mathbf{y}\right) - \mathbf{g}_{ij}\left(\mathbf{x}\right)\right] \tag{C.4}$$

Substituting (C.3) gives

$$\begin{aligned}\frac{\partial \mathbf{g}_{ij}}{\partial \mathbf{x}} = \lim_{\theta\to 0}\frac{1}{\theta}\Bigg[&k_{ij}\left(\|(\mathbf{x}_j+\theta\mathbf{y}_j)-(\mathbf{x}_i+\theta\mathbf{y}_i)\|-r_{ij}^0\right)\frac{((\mathbf{x}_j+\theta\mathbf{y}_j)-(\mathbf{x}_i+\theta\mathbf{y}_i))}{\|(\mathbf{x}_j+\theta\mathbf{y}_j)-(\mathbf{x}_i+\theta\mathbf{y}_i)\|}\\ &-k_{ij}\left(\|\mathbf{x}_j-\mathbf{x}_i\|-r_{ij}^0\right)\frac{(\mathbf{x}_j-\mathbf{x}_i)}{\|\mathbf{x}_j-\mathbf{x}_i\|}\Bigg]\end{aligned} \tag{C.5}$$

Letting $\Delta\mathbf{x} = \mathbf{x}_j - \mathbf{x}_i$ and $\Delta\mathbf{y} = \mathbf{y}_j - \mathbf{y}_i$ and simplifying, then (C.5) becomes

$$\begin{aligned}\frac{\partial \mathbf{g}_{ij}}{\partial \mathbf{x}} = \lim_{\theta\to 0}\frac{k_{ij}}{\theta}&\left[\theta\Delta\mathbf{y}-\frac{\theta r_{ij}^0\Delta\mathbf{y}}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|}\right]\\ &+\lim_{\theta\to 0}\frac{k_{ij}r_{ij}^0\Delta\mathbf{x}}{\theta}\left[\frac{1}{\|\mathbf{x}\|}-\frac{1}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|}\right]\end{aligned} \tag{C.6}$$

Focusing on the second term of (C.6):

$$\frac{1}{\|\mathbf{x}\|}-\frac{1}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|}=\frac{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|-\|\Delta\mathbf{x}\|}{\|\Delta\mathbf{x}\|\,\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|} \tag{C.7}$$

Then, using the fact that the Euclidean norm is induced by an inner product and rearranging terms:

$$\frac{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|-\|\Delta\mathbf{x}\|}{\|\Delta\mathbf{x}\|\,\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|}=\frac{\sqrt{1+2\theta\frac{\Delta\mathbf{x}\cdot\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^2}+\theta^2\frac{\Delta\mathbf{y}\cdot\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^2}}-1}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|} \tag{C.8}$$

Now, let $z = 2\theta\frac{\Delta\mathbf{x}\cdot\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^2}+\theta^2\frac{\Delta\mathbf{y}\cdot\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^2}$. Using a Taylor expansion on $\sqrt{1+z}$ about 0 and keeping up to first order terms, yields

$$\frac{\partial \mathbf{g}_{ij}}{\partial \mathbf{x}} = k_{ij}\Delta\mathbf{y}-\frac{k_{ij}r_{ij}^0\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|}+\lim_{\theta\to 0}\frac{k_{ij}r_{ij}^0\Delta\mathbf{x}}{\theta\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|}\left[\theta\frac{\Delta\mathbf{x}\cdot\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^2}+\frac{\theta^2}{2}\frac{\Delta\mathbf{y}\cdot\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^2}\right] \tag{C.9}$$

Taking the limit and collecting terms finally yields

$$\boxed{\frac{\partial \mathbf{g}_{ij}(\Delta\mathbf{x})}{\partial \mathbf{x}}(\Delta\mathbf{y}) = k_{ij}\Delta\mathbf{y}-\frac{k_{ij}r_{ij}^0\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|}+\frac{k_{ij}r_{ij}^0\,(\Delta\mathbf{x}\cdot\Delta\mathbf{y})}{\|\Delta\mathbf{x}\|^3}\Delta\mathbf{x}} \tag{C.10}$$

and therefore

$$\boxed{\mathbf{H}_{ij} = k_{ij}\left(1-\frac{r_{ij}^0}{r_{ij}}\right)\mathbf{I}+\frac{k_{ij}r_{ij}^0}{r_{ij}^3}\Delta\mathbf{x}\otimes\Delta\mathbf{x}} \tag{C.11}$$

C.2 Lennard-Jones Potential

The Lennard-Jones potential is

$$E_{ij}^{LJ} = 4\varepsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] \tag{C.12}$$

where r_{ij} is the distance between particles i and j and ε_{ij} and σ_{ij} are parameters. Then,

$$\frac{\partial E_{ij}^{LJ}}{\partial r_{ij}} = \frac{-48\varepsilon_{ij}}{\sigma_{ij}}\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{13} + \frac{24\varepsilon_{ij}}{\sigma_{ij}}\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{7} \tag{C.13}$$

so that, using the notation from the previous section,

$$\boxed{\mathbf{g}_{ij}(\Delta\mathbf{x}) = \left[\frac{-48\varepsilon_{ij}\sigma_{ij}^{12}}{\|\Delta\mathbf{x}\|^{13}} + \frac{24\varepsilon_{ij}\sigma_{ij}^{6}}{\|\Delta\mathbf{x}\|^{7}}\right]\frac{\Delta\mathbf{x}}{\|\Delta\mathbf{x}\|}} \tag{C.14}$$

Then,

$$\begin{aligned}\mathbf{H}_{ij} = \frac{\partial \mathbf{g}_{ij}(\Delta\mathbf{x})}{\partial \mathbf{x}}(\Delta\mathbf{y}) = \lim_{\theta\to 0}\frac{1}{\theta}\Bigg\{&\left[\frac{-48\varepsilon_{ij}\sigma_{ij}^{12}}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{13}} + \frac{24\varepsilon_{ij}\sigma_{ij}^{6}}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{7}}\right]\frac{\Delta\mathbf{x}+\theta\Delta\mathbf{y}}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|}\\ &- \left[\frac{-48\varepsilon_{ij}\sigma_{ij}^{12}}{\|\Delta\mathbf{x}\|^{13}} + \frac{24\varepsilon_{ij}\sigma_{ij}^{6}}{\|\Delta\mathbf{x}\|^{7}}\right]\frac{\Delta\mathbf{x}}{\|\Delta\mathbf{x}\|}\Bigg\}\end{aligned} \tag{C.15}$$

Rearranging terms gives

$$\begin{aligned}\frac{\partial \mathbf{g}_{ij}(\Delta\mathbf{x})}{\partial \mathbf{x}}(\Delta\mathbf{y}) =&\lim_{\theta\to 0}\frac{1}{\theta}\left[\theta\left(\frac{24\varepsilon_{ij}\sigma_{ij}^{6}\Delta\mathbf{y}}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{8}} - \frac{48\varepsilon_{ij}\sigma_{ij}^{12}\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^{14}}\right)\right]\\ &+ \lim_{\theta\to 0}\frac{1}{\theta}\left[48\varepsilon_{ij}\sigma_{ij}^{12}\Delta\mathbf{x}\left(\frac{1}{\|\Delta\mathbf{x}\|^{14}} - \frac{1}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{14}}\right)\right]\\ &+ \lim_{\theta\to 0}\frac{1}{\theta}\left[24\varepsilon_{ij}\sigma_{ij}^{6}\Delta\mathbf{x}\left(\frac{1}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{8}} - \frac{1}{\|\Delta\mathbf{x}\|^{8}}\right)\right]\end{aligned} \tag{C.16}$$

Applying the steps used earlier to obtain (C.9) to the second and third terms of (C.16) leads to

$$\begin{aligned}\frac{\partial \mathbf{g}_{ij}(\Delta\mathbf{x})}{\partial \mathbf{x}}(\Delta\mathbf{y}) = \lim_{\theta\to 0}\frac{1}{\theta}\Bigg\{\theta\left(\frac{24\varepsilon_{ij}\sigma_{ij}^{6}\Delta\mathbf{y}}{\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{8}} - \frac{48\varepsilon_{ij}\sigma_{ij}^{12}\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^{14}}\right)\\ +48\varepsilon_{ij}\sigma_{ij}^{12}\Delta\mathbf{x}\left(\frac{14\theta(\Delta\mathbf{x}\cdot\Delta\mathbf{y})+7\theta^{2}(\Delta\mathbf{y}\cdot\Delta\mathbf{y})}{\|\Delta\mathbf{x}\|^{2}\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{14}}\right)\\ -24\varepsilon_{ij}\sigma_{ij}^{6}\Delta\mathbf{x}\left(\frac{8\theta(\Delta\mathbf{x}\cdot\Delta\mathbf{y})+4\theta^{2}(\Delta\mathbf{y}\cdot\Delta\mathbf{y})}{\|\Delta\mathbf{x}\|^{2}\|\Delta\mathbf{x}+\theta\Delta\mathbf{y}\|^{8}}\right)\Bigg\}\end{aligned} \qquad \text{(C.17)}$$

Finally, taking the limit yields

$$\boxed{\begin{aligned}\frac{\partial \mathbf{g}_{ij}(\Delta\mathbf{x})}{\partial \mathbf{x}}(\Delta\mathbf{y}) =&\frac{24\varepsilon_{ij}\sigma_{ij}^{6}\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^{8}} - \frac{48\varepsilon_{ij}\sigma_{ij}^{12}\Delta\mathbf{y}}{\|\Delta\mathbf{x}\|^{14}} + \frac{672\varepsilon_{ij}\sigma_{ij}^{12}(\Delta\mathbf{x}\cdot\Delta\mathbf{y})}{\|\Delta\mathbf{x}\|^{16}}\Delta\mathbf{x}\\ &-\frac{192\varepsilon_{ij}\sigma_{ij}^{6}(\Delta\mathbf{x}\cdot\Delta\mathbf{y})}{\|\Delta\mathbf{x}\|^{10}}\Delta\mathbf{x}\end{aligned}} \qquad \text{(C.18)}$$

and therefore

$$\boxed{\mathbf{H}_{ij} = \left(\frac{24\varepsilon_{ij}\sigma_{ij}^{6}}{r_{ij}^{8}} - \frac{48\varepsilon_{ij}\sigma_{ij}^{12}}{r_{ij}^{14}}\right)\mathbf{I} + \left(\frac{672\varepsilon_{ij}\sigma_{ij}^{12}}{r_{ij}^{16}} - \frac{192\varepsilon_{ij}\sigma_{ij}^{6}}{r_{ij}^{10}}\right)\Delta\mathbf{x}\otimes\Delta\mathbf{x}} \qquad \text{(C.19)}$$

Appendix D

Analytical Solution of Axial Extension Problem

Here, the exact solution for the problem of axial extension with lateral contraction is derived for the Compressible Neo-Hookean and the Compressible Mooney-Rivlin materials. The strategy is to prescribe the deformation on a unit cube that corresponds to uniform axial loading with lateral contraction, but to allow the deformation variables to serve as parameters. Then, knowing the force on the lateral plane must be zero, a nonlinear equation results that can be solved analytically for the axial deformation in terms of the lateral contraction. With this information, the uniaxial force that would deform the body in this way can be computed.

Notation and Problem Setup

Let the material body under consideration be a cube of size L_0. Furthermore, let the body be deformed under a prescribed motion such that its length in the x-direction becomes L and its width and thickness become w. See Figure D.1. This corresponds directly to an uniform axial loading positive x-plane.

The map for such a deformation is

$$\varphi = \begin{bmatrix} X\frac{L}{L_0} \\ Y\frac{w}{L_0} \\ Z\frac{w}{L_0} \end{bmatrix} \tag{D.1}$$

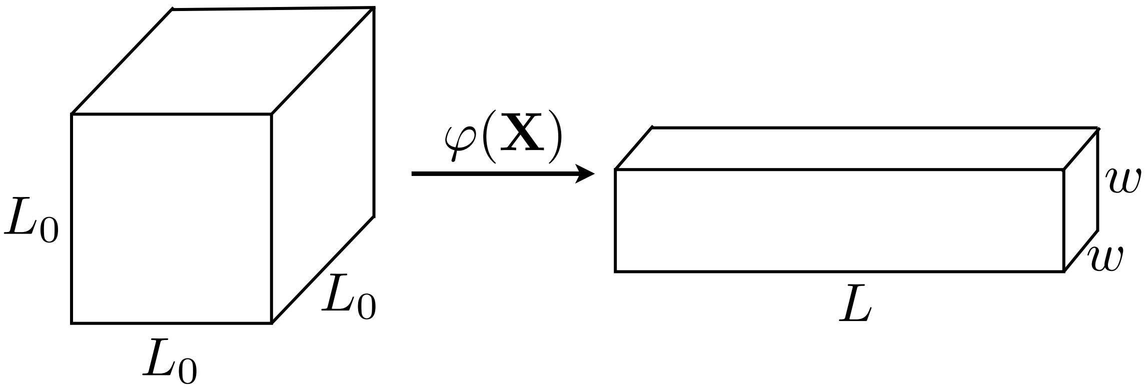


Figure D.1: Uniaxial extension with lateral contraction of a homogenous isotropic body.

Letting $\lambda_1 = L/L_0$ and $\lambda_2 = w/L_0$, then the deformation gradient $\mathbf{F}$ is

$$\nabla\varphi = \mathbf{F} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_2 \end{bmatrix} \tag{D.2}$$

and, therefore, the right Cauchy-Green strain tensor $\mathbf{C}$ is

$$\mathbf{C} = \mathbf{F}^T\mathbf{F} = \begin{bmatrix} \lambda_1^2 & 0 & 0 \\ 0 & \lambda_2^2 & 0 \\ 0 & 0 & \lambda_2^2 \end{bmatrix} \tag{D.3}$$

The invariants of the deformation are then

$$\begin{aligned} I_1 &= \text{tr } \mathbf{C} = \lambda_1^2 + 2\lambda_2^2 \\ I_2 &= \frac{1}{2}\left((\text{tr}\mathbf{C})^2 - \text{tr}\mathbf{C}^2\right) = \lambda_2^4 + 2\lambda_1^2\lambda_2^2 \\ I_3 &= \det \mathbf{C} = \lambda_1^2\lambda_2^4 \end{aligned} \tag{D.4}$$

From Appendix E, it is known that

$$\mathbf{S} = 2\left[\frac{\partial W}{\partial I_1}\mathbf{I} + \frac{\partial W}{\partial I_2}\left(I_1\mathbf{I} - \mathbf{C}\right) + \frac{\partial W}{\partial I_3}\left(I_3\mathbf{C}^{-1}\right)\right] \tag{D.5}$$

where W is the strain energy density function of the material body. Thus, for the uniaxial extension problem,

$$\boxed{\begin{aligned}\mathbf{S} =& 2\frac{\partial W}{\partial I_1}\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix} + 2\frac{\partial W}{\partial I_2}\begin{bmatrix}2\lambda_2^2 & 0 & 0\\ 0 & \lambda_1^2+\lambda_2^2 & 0\\ 0 & 0 & \lambda_1^2+\lambda_2^2\end{bmatrix}\\ &+2\frac{\partial W}{\partial I_3}\begin{bmatrix}\lambda_2^4 & 0 & 0\\ 0 & \lambda_1^2\lambda_2^2 & 0\\ 0 & 0 & \lambda_1^2\lambda_2^2\end{bmatrix}\end{aligned}} \tag{D.6}$$

Compressible Neo-Hookean Material

The strain energy density function for a compressible Neo-Hookean material is

$$W = C_1(I_1 - 3) + C_2(\sqrt{I_3} - 1)^2 - 2C_2 \ln \sqrt{I_3} \tag{D.7}$$

Therefore, differentiating and substituting expressions (D.4) yields

$$\frac{\partial W}{\partial I_1} = C_1, \frac{\partial W}{\partial I_2} = 0, \frac{\partial W}{\partial I_3} = -C_1\frac{1}{\lambda_1^2\lambda_2^4} + C_2\left(1 - \frac{1}{\lambda_1\lambda_2^2}\right) \tag{D.8}$$

Then, substituting into (D.6) gives

$$\mathbf{S} = \begin{bmatrix} \begin{array}{c} 2C_1\left(1-\dfrac{1}{\lambda_1^2}\right)+ \\ 2C_2\lambda_2^2\left(\lambda_2^2-\dfrac{1}{\lambda_1}\right)\end{array} & 0 & 0 \\ 0 & \begin{array}{c} 2C_1\left(1-\dfrac{1}{\lambda_2^2}\right)+ \\ 2C_2\lambda_1\left(\lambda_2^2\lambda_1-1\right)\end{array} & 0 \\ 0 & 0 & \begin{array}{c} 2C_1\left(1-\dfrac{1}{\lambda_2^2}\right)+ \\ 2C_2\lambda_1\left(\lambda_2^2\lambda_1-1\right)\end{array} \end{bmatrix} \tag{D.9}$$

Using the fact that the first Piola-Kirchhoff stress tensor $\mathbf{P} = \mathbf{F}\mathbf{S}$, then

$$\mathbf{P} = \begin{bmatrix} \begin{array}{l} 2C_1\left(\lambda_1 - \dfrac{1}{\lambda_1}\right) + \\ \quad 2C_2\lambda_2^2\left(\lambda_2^2\lambda_1 - 1\right) \end{array} & 0 & 0 \\ 0 & \begin{array}{l} 2C_1\left(\lambda_2 - \dfrac{1}{\lambda_2}\right) + \\ \quad 2C_2\lambda_1\lambda_2\left(\lambda_2^2\lambda_1 - 1\right) \end{array} & 0 \\ 0 & 0 & \begin{array}{l} 2C_1\left(\lambda_2 - \dfrac{1}{\lambda_2}\right) + \\ \quad 2C_2\lambda_1\lambda_2\left(\lambda_2^2\lambda_1 - 1\right) \end{array} \end{bmatrix} \tag{D.10}$$

It is known *a priori* that the force in the y-direction must be zero. Thus,

$$\mathbf{F}_y = \int_{A_0} \mathbf{P}\mathbf{n}_y \, dA_0 = 0 \tag{D.11}$$

where $\mathbf{n}_y = [0, 1, 0]^T$. Then,

$$\begin{aligned} 0 &= 2C_1\left(\lambda_2 - \frac{1}{\lambda_2}\right) + 2C_2\lambda_1\lambda_2\left(\lambda_2^2\lambda_1 - 1\right) \\ &= \left(C_2\lambda_2^3\right)\lambda_1^2 - \left(C_2\lambda_2\right)\lambda_1 + C_1\left(\lambda_2 - \frac{1}{\lambda_2}\right) \end{aligned} \tag{D.12}$$

Using the quadratic formula and simplifying, then λ_1 can be written as a function of λ_2:

$$\boxed{\lambda_1 = \frac{1 \pm \sqrt{1 + \frac{4C_1}{C_2}\left(1 - \lambda_2^2\right)}}{2\lambda_2^2}} \tag{D.13}$$

Furthermore, the force on the positive x-plane

$$\mathbf{F}_x = \int_{A_0} \mathbf{P}\mathbf{n}_x \, dA_0 \tag{D.14}$$

where $\mathbf{n}_x = [1, 0, 0]^T$. Thus, the applied traction (in the x-direction) that corresponds to this deformation is

$$\boxed{t_x = 2C_1\left(\lambda_1 - \frac{1}{\lambda_1}\right) + 2C_2\lambda_2^2\left(\lambda_2^2\lambda_1 - 1\right)} \tag{D.15}$$

Compressible Mooney-Rivlin Material

The strain energy density function for a compressible Mooney-Rivlin material is

$$W = C_1(I_1 - 3) + C_2(I_2 - 3) + C_3(\sqrt{I_3} - 1)^2 - (2C_1 + 4C_2)\ln\sqrt{I_3} \tag{D.16}$$

Differentiating with respect to the invariants gives

$$\frac{\partial W}{\partial I_1} = C_1, \frac{\partial W}{\partial I_2} = C_2, \frac{\partial W}{\partial I_3} = C_3\left(1 - \frac{1}{\lambda_1\lambda_2^2}\right) - (C_1 + 2C_2)\frac{1}{\lambda_1^2\lambda_2^4} \tag{D.17}$$

Repeating the same procedure used for the previous material, the second Piola-Kirchhoff stress for this material is

$$\mathbf{S} = \begin{bmatrix} \begin{array}{l} 2C_1\left(1 - \frac{1}{\lambda_1^2}\right) + \\ 4C_2\left(\lambda_2^2 - \frac{1}{\lambda_1^2}\right) + \\ 2C_3\lambda_2^2\left(\lambda_2^2 - \frac{1}{\lambda_1}\right) \end{array} & 0 & 0 \\ 0 & \begin{array}{l} 2C_1\left(1 - \frac{1}{\lambda_2^2}\right) + \\ 2C_2\left(\lambda_1^2 + \lambda_2^2 - \frac{2}{\lambda_2^2}\right) + \\ 2C_3\lambda_1\left(\lambda_1\lambda_2^2 - 1\right) \end{array} & 0 \\ 0 & 0 & \begin{array}{l} 2C_1\left(1 - \frac{1}{\lambda_2^2}\right) + \\ 2C_2\left(\lambda_1^2 + \lambda_2^2 - \frac{2}{\lambda_2^2}\right) + \\ 2C_3\lambda_1\left(\lambda_1\lambda_2^2 - 1\right) \end{array} \end{bmatrix} \tag{D.18}$$

and the first Piola-Kirchhoff stress tensor is

$$\mathbf{P} = \begin{bmatrix} \begin{array}{l} 2C_1\left(\lambda_1 - \dfrac{1}{\lambda_1}\right) + \\ 4C_2\lambda_1\left(\lambda_2^2 - \dfrac{1}{\lambda_1^2}\right) + \\ 2C_3\lambda_2^2\left(\lambda_1\lambda_2^2 - 1\right) \end{array} & 0 & 0 \\ 0 & \begin{array}{l} 2C_1\left(\lambda_2 - \dfrac{1}{\lambda_2}\right) + \\ 2C_2\lambda_2\left(\lambda_1^2 + \lambda_2^2 - \dfrac{2}{\lambda_2^2}\right) + \\ 2C_3\lambda_1\lambda_2\left(\lambda_1\lambda_2^2 - 1\right) \end{array} & 0 \\ 0 & 0 & \begin{array}{l} 2C_1\left(\lambda_2 - \dfrac{1}{\lambda_2}\right) + \\ 2C_2\lambda_2\left(\lambda_1^2 + \lambda_2^2 - \dfrac{2}{\lambda_2^2}\right) + \\ 2C_3\lambda_1\lambda_2\left(\lambda_1\lambda_2^2 - 1\right) \end{array} \end{bmatrix} \tag{D.19}$$

Noting that the traction vector in the y-direction must be zero, then

$$\begin{aligned} 0 &= 2C_1\left(\lambda_2 - \frac{1}{\lambda_2}\right) + 2C_2\lambda_2\left(\lambda_1^2 + \lambda_2^2 - \frac{2}{\lambda_2^2}\right) + 2C_3\lambda_1\lambda_2\left(\lambda_1\lambda_2^2 - 1\right) \\ &= \left(C_2\lambda_2 + C_3\lambda_2^3\right)\lambda_1^2 - \left(C_3\lambda_2\right)\lambda_1 + C_1\left(\lambda_2 - \frac{1}{\lambda_2}\right) + C_2\left(\lambda_2^3 - \frac{2}{\lambda_2}\right) \end{aligned} \tag{D.20}$$

Using the quadratic equation as before, then the solution is given by

$$\boxed{\lambda_1 = \frac{1 \pm \sqrt{1 - 4\left(\frac{C_2}{C_3^2\lambda_2} + \frac{\lambda_2}{C_3}\right)\left[C_1\left(\lambda_2 - \frac{1}{\lambda_2}\right) + C_2\left(\lambda_2^3 - \frac{2}{\lambda_2}\right)\right]}}{2\left(\frac{C_2}{C_3} + \lambda_2^2\right)}} \tag{D.21}$$

and the applied traction is

$$\boxed{t_x = 2C_1\left(\lambda_1 - \frac{1}{\lambda_1}\right) + 4C_2\lambda_1\left(\lambda_2^2 - \frac{1}{\lambda_1^2}\right) + 2C_3\lambda_2^2\left(\lambda_1\lambda_2^2 - 1\right)} \tag{D.22}$$

Appendix E

Derivation of Equations for Nonlinear Elasticity

Derivation of Discrete Gradient

Let $\varphi_i = \sum_C d_{\underline{k}C} N_C^{\underline{k}}$, where $N_C^{\underline{k}}$ represents the basis function of global node number C in the coordinate direction k and $d_{\underline{k}C}$ is the weighting coefficient. Note that, while in practice the same basis functions will be used in each direction, labeling the coordinate directions makes the presentation more clear. Thus

$$\begin{aligned}\Pi(d_{kC}) = &\int_{\Omega_0} W(\nabla(\sum_C d_{\underline{k}C} N_C^{\underline{k}}))d\mathbf{X} - \int_{\Omega_0} \rho_0 B_k(\sum_C d_{\underline{k}C} N_C^{\underline{k}})d\mathbf{X} - \\ &\int_{\Gamma_N} T_k(\sum_C d_{\underline{k}C} N_C^{\underline{k}})d\Gamma_N\end{aligned} \tag{E.1}$$

Note that in the second and third terms, the k index is summed on the $\underline{k}$ index (although the $\underline{k}$ indices are not summed relative to one another).

Thus,

$$\frac{\partial \Pi(d_{kC})}{\partial d_{iA}} = \int_{\Omega_0} \frac{\partial W}{\partial E_{IJ}} \frac{\partial E_{IJ}}{\partial d_{iA}} d\mathbf{X} - \int_{\Omega_0} \rho_0 B_{\underline{i}} N_A^{\underline{i}} d\mathbf{X} - \int_{\Gamma_N} T_{\underline{i}} N_A^{\underline{i}} d\Gamma_N \tag{E.2}$$

First, note that, by definition, $\partial W / \partial E_{IJ} = S_{IJ}$ where S_{IJ} is the second Piola-Kirchhoff stress. Next, $\partial E_{IJ} / \partial d_{iA}$ must be computed. By definition,

$$E_{IJ} = \frac{1}{2}\left(F_{iI} F_{iJ} - \delta_{IJ}\right) \tag{E.3}$$

where δ_{IJ} is the Kronecker delta. Thus,

$$\frac{\partial E_{ij}}{\partial d_{iA}} = \frac{1}{2}\left(\frac{dF_{iI}}{dd_{iA}}F_{iJ} + F_{iI}\frac{dF_{iJ}}{dd_{iA}}\right) \tag{E.4}$$

Now, by definition,

$$F_{kI} = \varphi_{k,I} = \sum_{C} d_{\underline{k}C} N^{\underline{k}}_{C,I} \tag{E.5}$$

Thus,

$$\frac{dF_{iI}}{dd_{iA}} = N^{i}_{A,I} \tag{E.6}$$

so that

$$\frac{\partial E_{ij}}{\partial d_{iA}} = \frac{1}{2}\left(N^{\underline{i}}_{A,I}F_{\underline{i}J} + F_{\underline{i}I}N^{\underline{i}}_{A,J}\right) \tag{E.7}$$

Substituting (E.7) into (E.2) and using the fact that $S_{IJ} = S_{JI}$, then

$$\boxed{\frac{\partial \Pi(d_{kC})}{\partial d_{iA}} = \int_{\Omega_0} F_{\underline{i}I}S_{IJ}N^{\underline{i}}_{A,J}d\mathbf{X} - \int_{\Omega_0} \rho_0 B_{\underline{i}}N^{\underline{i}}_{A}d\mathbf{X} - \int_{\Gamma_N} T_{\underline{i}}N^{\underline{i}}_{A}d\Gamma_N} \tag{E.8}$$

Derivation of Discrete Hessian

$$\frac{\partial^2 \Pi}{\partial d_{iA}\partial d_{jB}} = \int_{\Omega_0} \frac{dF_{\underline{i}I}}{dd_{jB}}S_{IJ}N^{\underline{i}}_{A,J} + F_{\underline{i}I}\frac{\partial S_{IJ}}{\partial E_{KL}}\frac{\partial E_{KL}}{\partial d_{jB}}N^{\underline{i}}_{A,J}d\mathbf{X} \tag{E.9}$$

Using (E.7) and (E.6), letting $\partial S_{IJ}/\partial E_{KL} = C_{IJKL}$ and then substituting into (E.9), then

$$\begin{aligned}\frac{\partial^2 \Pi}{\partial d_{iA}\partial d_{jB}} = \int_{\Omega_0} & N^{\underline{j}}_{B,I}S_{IJ}N^{\underline{i}}_{A,J} + \\ & F_{\underline{i}I}C_{IJKL}\frac{1}{2}\left(N^{\underline{j}}_{B,K}F_{\underline{j}L} + N^{\underline{j}}_{B,L}F_{\underline{j}K}\right)N^{\underline{i}}_{A,J}d\mathbf{X}\end{aligned} \tag{E.10}$$

Since $C_{IJKL} = C_{IJLK}$, then finally

$$\boxed{\frac{\partial^2 \Pi}{\partial d_{iA}\partial d_{jB}} = \int_{\Omega_0} N^{\underline{j}}_{B,I}S_{IJ}N^{\underline{i}}_{A,J} + F_{\underline{i}I}C_{IJKL}F_{\underline{j}K}N^{\underline{i}}_{A,J}N^{\underline{j}}_{B,L}d\mathbf{X}} \tag{E.11}$$

Derivation of S_{IJ} and C_{IJKL} for a General $W(I_1, I_2, I_3)$

Assuming the existence of $W(I_1, I_2, I_3)$, then the forms of S_{IJ} and C_{IJKL} must be derived, where I_1, I_2, I_3 are the invariants of the right Cauchy-Green deformation tensor, $\boldsymbol{C}$.

By the chain rule and the definition $E_{IJ} = 1/2(C_{IJ} - \delta_{IJ})$,

$$S_{IJ} = \frac{\partial W}{\partial E_{IJ}} = \frac{\partial W}{\partial C_{IJ}} \frac{\partial C_{IJ}}{\partial E_{IJ}} = 2\frac{\partial W}{\partial C_{IJ}} \tag{E.12}$$

Similarly,

$$C_{IJKL} = \frac{\partial^2 W}{\partial E_{IJ} \partial E_{KL}} = 4\frac{\partial^2 W}{\partial C_{IJ} \partial C_{KL}} \tag{E.13}$$

Thus, it remains to compute $\partial W/\partial C_{IJ}$ and $\partial^2 W/\partial C_{IJ}\partial C_{KL}$.

Assuming $W = W(I_1, I_2, I_3)$ then, by the chain rule,

$$\frac{\partial W}{\partial C_{IJ}} = \frac{\partial W}{\partial I_1}\frac{\partial I_1}{\partial C_{IJ}} + \frac{\partial W}{\partial I_2}\frac{\partial I_2}{\partial C_{IJ}} + \frac{\partial W}{\partial I_3}\frac{\partial I_3}{\partial C_{IJ}} \tag{E.14}$$

Similarly,

$$\begin{aligned}\frac{\partial^2 W}{\partial C_{IJ}\partial C_{KL}} =& \frac{\partial^2 W}{\partial I_1^2}\frac{\partial I_1}{\partial C_{IJ}}\frac{\partial I_1}{\partial C_{KL}} + \frac{\partial^2 W}{\partial I_1 \partial I_2}\frac{\partial I_1}{\partial C_{IJ}}\frac{\partial I_2}{\partial C_{KL}} + \\ & \frac{\partial^2 W}{\partial I_1 \partial I_3}\frac{\partial I_1}{\partial C_{IJ}}\frac{\partial I_3}{\partial C_{KL}} + \frac{\partial^2 W}{\partial I_2 \partial I_1}\frac{\partial I_2}{\partial C_{IJ}}\frac{\partial I_1}{\partial C_{KL}} + \\ & \frac{\partial^2 W}{\partial I_2^2}\frac{\partial I_2}{\partial C_{IJ}}\frac{\partial I_2}{\partial C_{KL}} + \frac{\partial^2 W}{\partial I_2 \partial I_3}\frac{\partial I_2}{\partial C_{IJ}}\frac{\partial I_3}{\partial C_{KL}} + \\ & \frac{\partial^2 W}{\partial I_3 \partial I_1}\frac{\partial I_3}{\partial C_{IJ}}\frac{\partial I_1}{\partial C_{KL}} + \frac{\partial^2 W}{\partial I_3 \partial I_2}\frac{\partial I_3}{\partial C_{IJ}}\frac{\partial I_2}{\partial C_{KL}} + \\ & \frac{\partial^2 W}{\partial I_3^2}\frac{\partial I_3}{\partial C_{IJ}}\frac{\partial I_3}{\partial C_{KL}} + \\ & \frac{\partial W}{\partial I_1}\frac{\partial^2 I_1}{\partial C_{IJ}\partial C_{KL}} + \frac{\partial W}{\partial I_2}\frac{\partial^2 I_2}{\partial C_{IJ}\partial C_{KL}} + \frac{\partial W}{\partial I_3}\frac{\partial^2 I_3}{\partial C_{IJ}\partial C_{KL}}\end{aligned} \tag{E.15}$$

Now:

$$\begin{aligned}
\frac{\partial I_1}{\partial C_{IJ}} &= \delta_{IJ},\ \frac{\partial I_2}{\partial C_{IJ}} = I_1\delta_{IJ} - C_{IJ},\ \frac{\partial I_3}{\partial C_{IJ}} = I_3 C_{IJ}^{-1} \\
\frac{\partial^2 I_1}{\partial C_{IJ}\partial C_{KL}} &= 0 \\
\frac{\partial^2 I_2}{\partial C_{IJ}\partial C_{KL}} &= \delta_{IJ}\delta_{KL} - \frac{1}{2}\left(\delta_{IK}\delta_{JL} + \delta_{IL}\delta_{JK}\right) \\
\frac{\partial^2 I_3}{\partial C_{IJ}\partial C_{KL}} &= I_3 C_{IJ}^{-1} C_{KL}^{-1} - \frac{I_3}{2}\left(C_{IK}^{-1}C_{JL}^{-1} + C_{IL}^{-1}C_{JK}^{-1}\right)
\end{aligned} \tag{E.16}$$

Thus,

$$\boxed{S_{IJ} = 2\left(\frac{\partial W}{\partial I_1}\delta_{IJ} + \frac{\partial W}{\partial I_2}\left(I_1\delta_{IJ} - C_{IJ}\right) + \frac{\partial W}{\partial I_3}\left(I_3 C_{IJ}^{-1}\right)\right)} \tag{E.17}$$

and

$$\boxed{\begin{aligned}
C_{IJKL} = 4\Bigg(&\frac{\partial^2 W}{\partial I_1^2}\left(\delta_{IJ}\delta_{KL}\right) + \\
&\frac{\partial^2 W}{\partial I_2^2}\left(I_1\delta_{IJ}\left(I_1\delta_{KL} - C_{KL}\right) + C_{IJ}\left(\delta_{KL} - C_{KL}\right)\right) + \\
&\frac{\partial^2 W}{\partial I_3^2}\left(I_3^2 C_{IJ}^{-1} C_{Kl}^{-1}\right) + \\
&\frac{\partial^2 W}{\partial I_1 \partial I_2}\left(I_1\delta_{IJ}\left(\delta_{KL} - C_{KL}\right) + I_1\delta_{KL}\left(\delta_{IJ} - C_{IJ}\right)\right) + \\
&\frac{\partial^2 W}{\partial I_1 \partial I_3} I_3\left(\delta_{IJ} C_{KL}^{-1} + C_{IJ}^{-1}\delta_{KL}\right) + \\
&\frac{\partial^2 W}{\partial I_2 \partial I_3} I_3\left(C_{KL}^{-1}\left(I_1\delta_{IJ} - C_{IJ}\right) + C_{IJ}^{-1}\left(I_1\delta_{KL} - C_{KL}\right)\right) + \\
&\frac{\partial W}{\partial I_2}\left(\delta_{IJ}\delta_{KL} - \delta_{IK}\delta JL\right) + \\
&\frac{\partial W}{\partial I_3} I_3\left(C_{IJ}^{-1}C_{KL}^{-1} - \frac{1}{2}\left(C_{IK}^{-1}C_{JL}^{-1} + C_{IL}^{-1}C_{JK}^{-1}\right)\right)\Bigg)
\end{aligned}} \tag{E.18}$$

Appendix F

Lemmas for 1-D Model Problem

F.1 Technical Lemmas

Here, the classical Poincaré inequality in one dimension is stated without proof:

Lemma F.1. ***(Poincaré Inequality)*** *Let $v \in H^1(\Omega_c)$. Then*

$$\|v\|^2_{L^2(\Omega_c)} \leq \frac{|\Omega_c|^2}{2}|v|^2_{H^1(\Omega_c)} \leq \frac{|\Omega_c|^2}{2E}\|v\|^2_{V_c} \tag{F.1}$$

Lemma F.2. *Let $v \in H^1(\Omega_o)$ and let $\bar{v}$ be the average of v on Ω_o, i.e.*

$$\bar{v} = \frac{1}{|\Omega_o|}\int_{\Omega_o} v dx$$

Then

$$|\Omega_o| \ \bar{v}^2 \leq \|v\|^2_{L^2(\Omega_o)} \leq |\Omega_o| \ \bar{v}^2 + \frac{|\Omega_o|^2}{2}|v|^2_{H^1(\Omega_o)}$$

Proof. Let $v \in H^1(\Omega_o)$. Note that

$$\begin{aligned}\int_{\Omega_o}(v-\bar{v})^2 dx &= \int_{\Omega_o} v^2 - 2v\bar{v} + \bar{v}^2 dx \\ &= \|v\|^2_{L^2(\Omega_o)} - 2\bar{v}\int_{\Omega_o} vdx + \bar{v}^2|\Omega_o| \\ &= \|v\|^2_{L^2(\Omega_o)} - \bar{v}^2|\Omega_o|\end{aligned}$$

The first inequality follows by observing that the integral on the left hand side is necessarily non-negative.

Let Ω_o be represented as the interval (x_a, x_b). Since v is continuous on Ω_o, then there exists $\bar{x}$, $x_a \leq \bar{x} \leq x_b$ such that $v(\bar{x}) = \bar{v}$. Decompose $v = \bar{v} + y$. Then

$$\begin{aligned}
\|v\|^2_{L^2(\Omega_o)} &= \int_{\Omega_o} (\bar{v} + y)^2 dx \\
&= \int_{\Omega_o} \bar{v}^2 dx + 2\bar{v} \int_{\Omega_o} y dx + \int_{\Omega_o} y^2 dx \\
&= \bar{v}^2 |\Omega_o| + \int_{\Omega_o} y^2 dx
\end{aligned}$$

as the average of y, by definition, is simply zero. Moreover, since y vanishes at $\bar{x}$ in Ω_o, and using the Poincaré inequality:

$$\int_{\Omega_o} y^2 dx \leq \frac{|\Omega_o|^2}{2} |v|^2_{H^1(\Omega_o)}$$

which completes the proof.

$\square$

Lemma F.3. *Let $z \in \mathbb{R}^{n_o+1}$ and let $\bar{z}$ be the average of z on Ω_o. Then*

$$z^2_{n_0} \leq 2\bar{z}^2 + 2\sum_{i=1}^{n_0} (z_i - z_{i-1})^2$$

Proof. Let $\bar{z}_i$, $i = 1, \ldots, n_o$ be defined as:

$$\bar{z}_i = \frac{l_i}{|\Omega_o|} \frac{z_i + z_{i-1}}{2}$$

Thus,

$$\bar{z}_i - \frac{l_i}{|\Omega_o|} \frac{z_i - z_{i-1}}{2} = \frac{l_i}{|\Omega_o|} z_{i-1} = \frac{l_i}{|\Omega_o|} \left(z_{n_0} - \sum_{k=i}^{n_0} (z_k - z_{k-1}) \right)$$

that is:

$$\frac{l_i}{|\Omega_o|} z_{n_0} = \bar{z}_i - \frac{l_i}{|\Omega_o|} \left(\frac{z_i - z_{i-1}}{2} - \sum_{k=i}^{n_0} (z_k - z_{n_0}) \right)$$

Summing over all terms in $i = 1, \ldots, n_0$, and noting that $\sum \bar{z}_i = \bar{z}$ and $\sum_i l_i = |\Omega_o|$, then

$$\begin{aligned}
z_{n_0} &= \bar{z} - \sum_{i=1}^{n_0} \frac{l_i}{|\Omega_o|} \left(\frac{z_i - z_{i-1}}{2} - \sum_{k=i}^{n_0} (z_k - z_{k-1}) \right) \\
&= \bar{z} + \sum_{i=1}^{n_0} \left[\frac{1}{|\Omega_o|} \left(\left(\sum_{k=1}^{i-1} l_k \right) + \frac{l_i}{2} \right) \right] (z_i - z_{i-1})
\end{aligned}$$

Therefore,

$$\begin{aligned}
|z_{n_0}| &\leq |\bar{z}| + \sum_{i=1}^{n_0} \left[\frac{1}{|\Omega_o|} \left(\left(\sum_{k=1}^{i-1} l_k \right) + \frac{l_i}{2} \right) \right] |z_i - z_{i-1}| \\
&\leq |\bar{z}| + \sum_{i=1}^{n_0} |z_i - z_{i-1}|
\end{aligned}$$

which yields the result, using the fact that $(a+b)^2 \leq 2(a^2+b^2)$ a, $b \in \mathbb{R}$.

□

F.2 Proof of Lemmas for the Continuous Problem

Lemma F.1 (Continuity of a). *Let $a(\cdot,\cdot)$ be the bilinear form defined in* (4.22). *Then, for all $U = (u, w)$, $V = (v, z) \in X$, there exists a constant $M_a > 0$ such that:*

$$|a(U,V)| \leq M_a \|U\|_X \|(V)\|_X$$

with $M_a = 1$.

Proof. The Cauchy-Schwarz and Hölder inequalities give

$$\begin{aligned}
|a(U,V)| &\leq \int_{\Omega_c} \alpha_c E |u'||v'| \, dx + \sum_{i=1}^{m} \alpha_i k_i |w_i - w_{i-1}||z_i - z_{i-1}| \\
&\leq C_1 \|u\|_{V_c} \|v\|_{V_c} + C_2 |w|_{V_d} |z|_{V_d}
\end{aligned}$$

where $C_1 = \max_x(\alpha_c) = 1$ and $C_2 = \max_i(\alpha_i) = 1$. The definition of the norm in V_d yields

$$|a(U,V)| \leq \|u\|_{V_c} \|v\|_{V_c} + \|w\|_{V_d} \|z\|_{V_d} \leq \|U\|_X \|V\|_X$$

and $M_a = 1$. $\square$

Lemma F.2 (Continuity of b)**.** *Let* $b(\cdot,\cdot)$ *be as defined in* (4.22)*. Then, for all* $\mu \in M$*,* $V = (v, z) \in X$*, there exists a constant* $M_b > 0$ *such that:*

$$|b(\mu, V)| \leq M_b \|\mu\|_M \|V\|_X$$

with

$$M_b = 2 \max \left(\sqrt{\frac{\beta_1 |\Omega_c|^2 + 2\beta_2}{2E}}, \sqrt{\frac{\beta_1}{\delta}|\Omega_o|}, \sqrt{\frac{\beta_1 |\Omega_o|^2 + 2\beta_2}{2 \min_i k_i l_i}} \right)$$

Proof. By making use of Poincaré inequality (F.1) and the fact that $(a+b)^2 \leq 2(a^2+b^2)$, $\forall a,\, b \in \mathbb{R}$, then

$$\begin{aligned} |b(\mu, V)| &\leq \|\mu\|_M \|v - \Pi z\|_M \\ &\leq \|\mu\|_M \left(\|v\|_M + \|\Pi z\|_M \right) \\ &\leq \sqrt{2} \|\mu\|_M \sqrt{\|v\|_M^2 + \|\Pi z\|_M^2} \end{aligned}$$

Now,

$$\begin{aligned} \|v\|_M^2 &= \beta_1 \|v\|_{L^2(\Omega_o)}^2 + \beta_2 |v|_{H^1(\Omega_o)}^2 \\ &\leq \beta_1 \|v\|_{L^2(\Omega_c)}^2 + \beta_2 |v|_{H^1(\Omega_c)}^2 \\ &\leq \frac{\beta_1 |\Omega_c|^2 + 2\beta_2}{2E} \|v\|_{V_c}^2 \end{aligned}$$

In the same way, using Lemma F.2 and the fact that Πz is a piecewise linear continuous function, then

$$\begin{aligned} \|\Pi z\|_M^2 &= \beta_1 \|\Pi z\|_{L^2(\Omega_o)}^2 + \beta_2 |\Pi z|_{H^1(\Omega_o)}^2 \\ &\leq \beta_1 |\Omega_o| \; \overline{z}^2 + \left(\beta_1 \frac{|\Omega_o|^2}{2} + \beta_2 \right) |\Pi z|_{H^1(\Omega_o)}^2 \\ &\leq \frac{\beta_1}{\delta} |\Omega_o| \; \delta \overline{z}^2 + \left(\frac{\beta_1 |\Omega_o|^2 + 2\beta_2}{2 \min_i k_i l_i} \right) |z|_{V_d}^2 \\ &\leq \max \left(\frac{\beta_1}{\delta} |\Omega_o|, \frac{\beta_1 |\Omega_o|^2 + 2\beta_2}{2 \min_i k_i l_i} \right) \|z\|_{V_d}^2 \end{aligned}$$

Combining the above results gives

$$|b(\mu, V)| \leq M_b \|\mu\|_M \|V\|_X$$

with:

$$M_b = 2\max\left(\sqrt{\frac{\beta_1|\Omega_c|^2 + 2\beta_2}{2E}}, \sqrt{\frac{\beta_1}{\delta}|\Omega_o|}, \sqrt{\frac{\beta_1|\Omega_o|^2 + 2\beta_2}{2\min_i k_i l_i}}\right)$$

$\square$

Lemma F.3 (Continuity of l). *Let $l(\cdot)$ be as defined in (4.22). Then, for all $V \in X$, there exists a constant $M_l > 0$ such that:*

$$|l\,(V)| \leq M_l \|V\|_X$$

with

$$M_l = 2|f|\max\left(\frac{1}{\sqrt{\delta}}, \frac{1}{\sqrt{\min_i k_i}}\right)$$

Proof. From definition of $l(\cdot)$, with $V = (v, z)$:

$$\begin{aligned} |l(V)| &\leq |f z_m| \leq |f||z_m| \leq |f|\left| z_{n_o} + \sum_{i=n_o+1}^{m}(z_i - z_{i-1})\right| \\ &\leq |f|\sqrt{2z_{n_o}^2 + 2\sum_{i=n_o+1}^{m}(z_i - z_{i-1})^2} \end{aligned}$$

Using Lemma F.3 yields:

$$\begin{aligned} |l(V)| &\leq |f|\sqrt{4\bar{z}^2 + 4\sum_{i=1}^{n_o}(z_i - z_{i-1})^2 + 2\sum_{i=n_o+1}^{m}(z_i - z_{i-1})^2} \\ &\leq |f|\sqrt{\frac{4}{\delta}\delta\bar{z}^2 + \frac{4}{\min_i k_i}\sum_{i=1}^{m}k_i(z_i - z_{i-1})^2} \\ &\leq 2|f|\sqrt{\max\left(\frac{1}{\delta}, \frac{1}{\min_i k_i}\right)(|z|_{V_d}^2 + \bar{z}^2)} \end{aligned}$$

It follows that

$$|l\,(V)| \leq M_l\|z\|_{V_d} \leq M_l\|(v, z)\|_X = M_l\|V\|_X$$

with

$$M_l = 2|f| \max\left(\frac{1}{\sqrt{\delta}}, \frac{1}{\sqrt{\min_i k_i}}\right)$$

□

Bibliography

[1] Simulation-based engineering science: Report by a blue ribbon panel to the national science foundation. May 2006. available at: http://www.ices.utexas.edu/events/SBES_Final_Report_070506.pdf.

[2] F. F. Abraham, J. Q. Broughton, N. Bernstein, and E. Kaxiras. Spanning the length scales in dynamic simulation. *Computers in Physics*, 12(6):538–546, 1998.

[3] M. Arndt and M. Luskin. Goal-oriented atomistic-continuum adaptivity for the quasicontinuum approximation. *Int. J. Multiscale Comput. Eng.*, 5:407–415, 2007.

[4] M. Arndt and M. Luskin. Goal-oriented adaptive mesh refinement for the quasicontinuum approximation of a frenkel-kontorova model. *Comput. Methods Appl. Mech. Engrg.*, 2008. To appear.

[5] I. Babuška. The finite element method with Lagrangian multipliers. *Numer. Math.*, 20:179–192, 1973.

[6] S. Badia, M. Parks, P. Bochev, M. Gunzburger, and R. Lehoucq. On atomistic-to-continuum (atc) coupling by blending. 2007. Sandia National Laboratories, Technical report SAND 2007-5126J.

[7] T. Bailey, S. Johnson, D. Resnick, S. Sreenivasan, J. Ekerdt, and C. Willson. Step and flash imprint lithography: An efficient nanoscale printing technology. *J. Photopolymer Sci. Tech.*, 15(9):481, 2002.

[8] S. Balay, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 2.1.5, Argonne National Laboratory, 2004.

[9] S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang. PETSc Web page, 2001. http://www.mcs.anl.gov/petsc.

[10] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith. Efficient management of parallelism in object oriented numerical software libraries. In E. Arge, A. M. Bruaset, and H. P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997.

[11] W. Bangerth and R. Rannacher. *Adaptive Finite Element Methods for Differential Equations*. Birkhäuser Verlag, 2003.

[12] P. T. Bauman, H. Ben Dhia, N. Elkhodja, J. T. Oden, and S. Prudhomme. On the application of the Arlequin method to the coupling of particle and continuum models. *Computational Mechanics*, 2008. To appear.

[13] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, 10:1–102, 2001.

[14] T. Belytschko and S. P. Xiao. Coupling methods for continuum model with molecular model. *International Journal for Multiscale Computational Engineering*, 1(1):115–126, 2003.

[15] H. Ben Dhia. Multiscale mechanical problems: the Arlequin method. *Comptes Rendus de l'Académie des Sciences Paris Série IIB*, 326(12):899–904, 1998.

[16] H. Ben Dhia and G. Rateau. Mathematical analysis of the mixed Arlequin method. *Comptes Rendus de l'Académie des Sciences Paris Série I*, 332:649–654, 2001.

[17] H. Ben Dhia and G. Rateau. Application of the Arlequin method to some structures with defects. *Revue Européenne des Eléments Finis*, 332:649–654, 2002.

[18] H. Ben Dhia and G. Rateau. The Arlequin method as a flexible engineering design tool. *Int. J. Numer. Meth. Engng.*, 62(11):1442–1462, 2005.

[19] S. Benson, L. C. McInnes, J. Moré, T. Munson, and J. Sarich. TAO user manual (revision 1.9). Technical Report ANL/MCS-TM-242, Mathematics and Computer Science Division, Argonne National Laboratory, 2007. http://www.mcs.anl.gov/tao.

[20] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers. *RAIRO Anal. Numér.*, pages 129–151, 1974.

[21] J. Q. Broughton, F. F. Abraham, N. Bernstein, and E. Kaxiras. Concurrent coupling of length scales: Methodology and application. *Physical Review B*, 60(4):2391–2403, 1999.

[22] W. H. Brown, C. S. Foote, and B. L. Iverson. *Organic Chemistry.* Brooks Cole, New York, 2004.

[23] R. L. Burns, S. C. Johnson, G. M. Schmid, E. K. Kim, M. D. Dickey, J. Meiring, S. D. Burns, N. A. Stacey, and C. G. Willson. Mesoscale modeling for sfil simulating polymerization kinetics and densification. *Proc. of SPIE*, 5374:348–360, 2004.

[24] P. G. Ciarlet. *Mathematical Elasticity Volume I: Three Dimensional Elasticity.* Elsevier, The Netherlands, 1988.

[25] E. Collister, 2008. Private Communication.

[26] W. A. Curtin and R. E. Miller. Atomistic/continuum coupling in computational materials science. *Modelling Simul. Mater. Sci. Eng.*, 11:R33–R68, 2003.

[27] M. D. Dickey, R. L. Burns, E. K. Kim, S. C. Johnson, N. A. Stacey, and C. G. Willson. Study of the kinetics of step and flash imprint lithography photopolymerization. *AIChE Journal*, 51(9):2547–2555, 2005.

[28] M. D. Dickey and C. G. Willson. Determination of the kinetic parameters for the step and flash imprint lithography photopolymerization. in press.

[29] L. M. Dupuy, E. B. Tadmor, R. E. Miller, and R. Phillips. Finite-temperature quasicontinuum: Molecular dynamics without all the atoms. *Physical Review Letters*, 5, 2005. Art. No. 060202.

[30] W. E and B. Engquist. The heterogeneous multiscale methods. *Comm. Math. Sci.*, 1(1):87–132, 2003.

[31] W. E, B. Engquist, X. Li, W. Ren, and E. Vanden-Eijnden. The heterogeneous multiscale method: A review, Preprint 2004. *Available at www.math.princeton.edu/multiscale/.*

[32] W. E, X. Li, and E. Vanden-Eijnden. Some recent progress in multiscale modeling, Preprint 2004. *Available at www.math.princeton.edu/multiscale/.*

[33] W. E, P. B. Ming, and P. W. Zhang. Analysis of the heterogeneous multiscale method for elliptic homogenization problems. *J. Amer. Math. Soc.*, 2004. to appear in print.

[34] A. Ern and J.-L. Guermond. *Theory and Practice of Finite Elements.* Springer-Verlag, New York, 2004.

[35] J. Fish, M. A. Nuggehally, M. S. Shephard, C. R. Picu, S. Badia, M. L. Parks, and M. Gunzburger. Concurrent atc coupling based on a blend of the continuum stress and the atomistic force. *Comput. Methods Appl. Mech. Engrg.*, 196:4548 – 4560, 2007.

[36] E. Giessen and A. Needleman. Discrete dislocation plasticity: a simple planar model. *Modelling Simul. Mater. Sci. Eng.*, 3:689–735, 1995.

[37] P. Guidault and T. Belytschko. On the L^2 and the H^1 couplings for an overlapping domain decomposition method using Lagrange multipliers. *Int. J. Numer. Meth. Engng.*, 70(3):322–350, 2007.

[38] W. Humphrey, A. Dalke, and K. Schulten. VMD – Visual Molecular Dynamics. *Journal of Molecular Graphics*, 14:33–38, 1996. Available at: http://www.ks.uiuc.edu/Research/vmd/.

[39] C. T. Kelley. *Iterative Methods for Linear and Nonlinear Equations.* SIAM, Philadelphia, PA, 1995.

[40] K. J. Laidler. *Chemical Kinetics.* Harper and Row, New York, 1987.

[41] X. S. Li and J. W. Demmel. SuperLU_DIST: A scalable distributed-memory sparse direct solver for unsymmetric linear systems. *ACM Trans. Mathematical Software*, 29(2):110–140, June 2003.

[42] W. K. Liu, E. G. Karpov, S. Zhang, and H. S. Park. An introduction to computational nanomechanics and materials. *Comput. Methods Appl. Mech. Engng.*, 193:1529–1578, 2004.

[43] M. J. Madou. *Fundamentals of Microfabrication.* CRC Press, Boca Raton, FL, 2002.

[44] R. Miller, M. Ortiz, R. Phillips, V. Shenoy, and E. B. Tadmor. Quasicontinuum models of fracture and plasticity. *Eng. Fracture Mech.*, 61:427–444, 1998.

[45] R. E. Miller and E. B. Tadmor. The quasicontinuum method: Overview, applications, and current directions. *Journal of Computer-Aided Design*, 9:203–239, 2002.

[46] R. E. Miller and E. B. Tadmor. *QC Tutorial Guide Version 1.1*, May 2004. Available at www.qcmethod.com.

[47] R. E. Miller, E. B. Tadmor, R. Phillips, and M. Ortiz. Quasicontinuum simulation of fracture at the atomic scale. *Modeling Simul. Mater. Sci. Eng.*, 6:607–638, 1998.

[48] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, New York, 1999.

[49] J. T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Computers and Mathematics with Applications*, 41:735–756, 2001.

[50] J. T. Oden and S. Prudhomme. Estimation of modeling error in computational mechanics. *Journal of Computational Physics*, 182:496–515, 2002.

[51] J. T. Oden, S. Prudhomme, and P. Bauman. On the extension of goal-oriented error estimation and hierarchical modeling to discrete lattice models. *Comput. Methods Appl. Mech. Engrg.*, 194:3668–3688, 2005.

[52] J. T. Oden, S. Prudhomme, A. Romkes, and P. T. Bauman. Multi-scale modeling of physical phenomena: Adaptive control of models. *SIAM Journal on Scientific Computing*, 28:2359–2389, 2006.

[53] J. T. Oden and K. Vemaganti. Estimation of local modeling error and goal-oriented modeling of heterogeneous materials 1. Error estimates and adaptive algorithms. *Journal of Computational Physics*, 164:22–47, 2000.

[54] J. T. Oden, K. Vemaganti, and N. Moës. Hierarchical modeling of heterogeneous solids. *Comput. Methods Appl. Mech. Engng.*, 172:3–25, 1999.

[55] J. T. Oden and T. I. Zohdi. Analysis and adaptive modeling of highly heterogeneous elastic structures. *Comput. Methods Appl. Mech. Engng.*, 148:367–391, 1997.

[56] R. W. Ogden. *Non-linear Elastic Deformations.* Halstead Press, New York, 1984.

[57] G. L. Patrick. *Orgranic Chemistry, Instant Notes.* BIOS Scientific Publishers, Taylor and Francis Group, London, 2004.

[58] S. Prudhomme, P. T. Bauman, and J. T. Oden. Error control for molecular statics problems. *Int. J. Multiscale Comp. Engng.*, 4:647–662, 2006.

[59] S. Prudhomme and J. T. Oden. On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Comput. Methods Appl. Mech. Engng.*, 176:313–331, 1999.

[60] D. Qian, G. J. Wagner, and W. K. Liu. A multiscale projection method for the analysis of carbon nanotubes. *Comput. Methods Appl. Mech. Engng.*, 193:1603–1632, 2004.

[61] R. S. Rivlin and D. W. Saunders. Large elastic deformations of isotropic materials: VII experiments on the deformation of rubber. *Phil. Trans. roy. Soc. Lon. A*, 243:251–288, 1951.

[62] A. Romkes and J. T. Oden. Adaptive modeling of wave propagation in heterogeneous elastic solids. *Comput. Methods Appl. Mech. Engng.*, 193:539–559, 2004.

[63] A. Romkes, J. T. Oden, and K. Vemaganti. Multi-scale goal-oriented adaptive modeling of random heterogeneous materials. *Mechanics of Materials*, 38:859–872, 2006.

[64] V. B. Shenoy, R. Miller, E. B. Tadmor, R. Phillips, and M. Ortiz. Quasicontinuum models of interfacial structure and deformation. *Physical Review Letters*, 80:742–745, 1998.

[65] V. B. Shenoy, R. Miller, E. B. Tadmor, D. Rodney, R. Phillips, and M. Ortiz. An adaptive finite element approach to atomic-scale mechanics — the quasicontinuum method. *Journal of the Mechanics and Physics of Solids*, 47:611–642, 1999.

[66] L. E. Shilkrot, R. E. Miller, and W. A. Curtin. Multiscale plasticity modeling: Coupled atomistics and discrete dislocation mechanics. *Journal of the Mechanics and Physics of Solids*, 52:755–787, 2004.

[67] G. S. Smith, E. B. Tadmor, N. Bernstein, and E. Kaxiras. Multiscale simulations of silicon nanoindentation. *Acta. Mater.*, 49:4089–4101, 2001.

[68] T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM Journal on Numerical Analysis*, 20:626–637, 1983.

[69] E. B. Tadmor. *The Quasicontinuum Method.* PhD thesis, Brown University, 1996.

[70] E. B. Tadmor, R. Miller, R. Phillips, and M. Ortiz. Nanoindentation and incipient plasticity. *J. Mater. Res.*, 14:2233–2250, 1999.

[71] E. B. Tadmor, M. Ortiz, and R. Phillips. Quasicontinuum analysis of defects in solids. *Phil. Mag. A*, 73(6):1529–1563, 1996.

[72] L. R. G. Treloar. *The Physics of Rubber Elasticity.* Clarendon Press, Oxford, 1975.

[73] K. Vemaganti and J. T. Oden. Estimation of local modeling error and goal-oriented modeling of heterogeneous materials; part ii: A computational environment for adaptive modeling of heterogeneous elastic solids. *Comput. Meth. Appl. Mech. Engrg.*, 190:6089–6124, 2001.

[74] G. J. Wagner and W. K. Liu. Coupling of atomistic and continuum simulations using a bridging scale decomposition. *Journal of Computational Physics*, 190:249–274, 2003.

[75] S. P. Xiao and T. Belytschko. A bridging domain method for coupling continua with molecular dynamics. *Comput. Methods Appl. Mech. Engng.*, 193:1645–1669, 2004.

[76] T. I. Zohdi, J. T. Oden, and G. J. Rodin. Hierarchical modeling of heterogeneous bodies. *Comput. Methods Appl. Mech. Engng.*, 138:273–298, 1996.

Vita

Paul Thomas Bauman II was born in Houston, Texas on August 20, 1980, the son of Thomas Arthur Bauman Sr. and Sharon Elaine McClelland. Following graduation from The Woodlands High School in The Woodlands, TX in 1999, he enrolled at The University of Texas at Austin. He received the Bachelor of Science degree with honors in Aerospace Engineering in 2002. He immediately entered graduate school at The University of Texas at Austin in the Computational and Applied Mathematics program under the supervision of J. Tinsley Oden. In 2003, he received his Masters of Science in Computational and Applied Mathematics. On June 9, 2007, he married Sara Lisabeth Hatton in Austin, TX.

Permanent Address: 6636 W William Cannon Dr
APT 1022
Austin, TX 78735

This dissertation was typeset with LaTeX 2_ε[1] by the author.

---

[1] LaTeX 2_ε is an extension of LaTeX. LaTeX is a collection of macros for TeX. TeX is a trademark of the American Mathematical Society. The macros used in formatting this dissertation were written by Dinesh Das, Department of Computer Sciences, The University of Texas at Austin, and extended by Bert Kay, James A. Bednar, and Ayman El-Khashab.